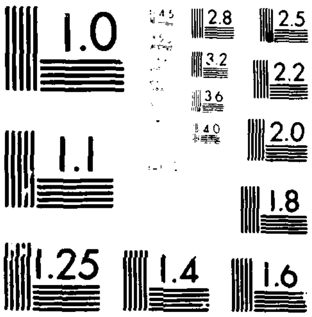
AD-A084 273 COLD REGIONS RESEARCH AND ENGINEERING LAB HANOVER NH F/G 8/12
DOCUMENTATION FOR A TWO-LEVEL DYNAMIC THERMODYNAMIC SEA ICE MOD--ETC(U)
FEB 80 W D HIBLER
UNCLASSIFIED CRREL-SR-80-8 NL

1 OF 1
AD
A084273

END
DATE
FILMED
6-80
DTIC



MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

12

SC

LEVEL II

Special Report 80-8

February 1980

ADA084273

DOCUMENTATION FOR A TWO-LEVEL DYNAMIC THERMODYNAMIC SEA ICE MODEL

William D. Hibler III

DTIC
ELECTE
MAY 20 1980
S D
E

Prepared for
OFFICE OF NAVAL RESEARCH
And
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
By

UNITED STATES ARMY
CORPS OF ENGINEERS
COLD REGIONS RESEARCH AND ENGINEERING LABORATORY
HANOVER, NEW HAMPSHIRE, U.S.A.

Approved for public release; distribution unlimited

DOC FIL

80 5 19 242

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
9 Special Report [redacted]

2. GOVT ACCESSION NO.
AD-A084 273

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
6 DOCUMENTATION FOR A TWO-LEVEL DYNAMIC THERMODYNAMIC SEA ICE MODEL

5. TYPE OF REPORT & PERIOD COVERED

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
10 William D. Hibler III

8. CONTRACT OR GRANT NUMBER(s)
Office of Naval Research
Requistion No. N0001479MP90003
NASA Requistion No. C-27021-D

9. PERFORMING ORGANIZATION NAME AND ADDRESS
U.S. Army Cold Regions Research and Engineering Laboratory
Hanover, New Hampshire 03755

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Office of Naval Research
Washington, D.C. 20360
National Aeronautics and Space Administration
Washington, D.C. 20546

12. ~~REPORT DATE~~
11 Febr[redacted]80

13. ~~NUMBER OF PAGES~~
38

14. MONITORING AGENCY NAME & ADDRESS(*if different from Controlling Office*)
12 38

15. SECURITY CLASS. (*of this report*)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (*of this Report*)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (*of the abstract entered in Block 20, if different from Report*)

14 CRREL-SR-8Ø-8

18. SUPPLEMENTARY NOTES

19. KEY WORDS (*Continue on reverse side if necessary and identify by block number*)
Arctic Ocean
Computerized simulation
Dynamics
Mathematical models
Sea ice
Thermodynamics

20. ABSTRACT (*Continue on reverse side if necessary and identify by block number*)

A discussion of the numerics and computer code for a two-level dynamic thermodynamic sea ice model is presented. For interested users a listing of the computer code and results from a 21-day test run are included as appendices. To a large degree this report is meant to serve as an extended appendix to an article by the author in the *Journal of Physical Oceanography* describing his model and a variety of simulation results. The model consists of a two-level ice thickness distribution coupled to the ice dynamics by a plastic rheology. In addition to the ice interaction the momentum balance includes nonlinear wind and water drag terms, Coriolis force, and inertial and momentum advection terms. The numerical scheme is formulated in an energy-conserving manner in a fixed Eulerian grid which allows simulation over unlimited time intervals. The momentum balance (including inertial terms) is numerically (over)

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

037100

20. Abstract (cont'd).

treated in a semi-implicit manner so that time steps of up to one day in length may be used if desired. The boundaries, grid size and time step magnitude are easily modified so that the model should have application to a variety of climate and forecasting problems.

Accession For
NTIS GRA&I ☑
DDC TAB □
Unannounced □
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or special
A

PREFACE

This report was prepared by William D. Hibler III, Research Physicist, of the Snow and Ice Branch, Research Division, U.S. Army Cold Regions Research and Engineering Laboratory. Funding for this research was provided by the Office of Naval Research, Code 461, Requistion No. N0001479MP90003, *Numerical Modeling of the Arctic Ice Cover and Ocean,* and by the National Aeronautics and Space Administration, Requisition No. C-27021-D, *Numerical Modeling of Sea Ice Dynamics and Ice Thickness Characteristics.*

The manuscript of this report was technically reviewed by W.B. Tucker and Dr. Miles McPhee of CRREL.

The contents of this report are not to be used for advertising or promotional purposes. Citation of brand names does not indicate an official endorsement or approval of the use of such commercial products.

DOCUMENTATION FOR A TWO-LEVEL DYNAMIC THERMODYNAMIC SEA ICE MODEL

William D. Hibler III

INTRODUCTION

In a recent article Hibler (1979) formulated and tested a two-level numerical model for the simulation of sea ice circulation and thickness over a seasonal cycle. Although developed for long-term simulations, this model contains many features which may prove useful in sea ice forecasting. In addition the code may well have utility in studies of ice covered lakes.

The purpose of this report is to document the numerical procedure and computer code used in this model. To aid in using this code, a complete listing and a printout of a double precision test run are attached as Appendices.

In this documentation frequent reference will be made to the article in the *Journal of Physical Oceanography* that describes the model (Hibler 1979 hereafter referred to as JPO79). In addition to describing a variety of simulation results, this article contains a complete description of the model equations and a discussion of the numerical scheme and finite difference code. Consequently, various equations in JPO79 will frequently be directly referred to by number in the text of this report.

BRIEF DESCRIPTION OF MODEL

Model structure

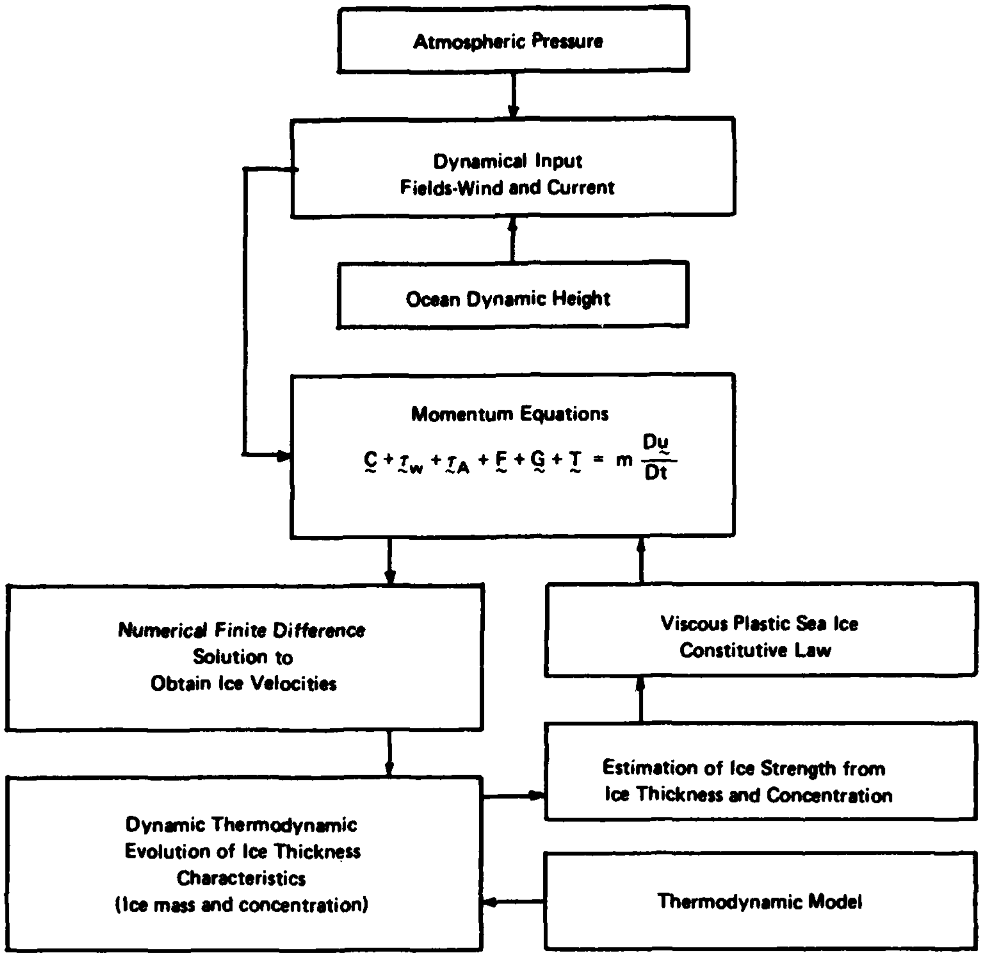
The overall structure (see Fig. 1) essentially consists of three main components. The first is *momentum balance* which includes air and water stress, Coriolis force, internal ice stress, inertial forces and ocean tilt (sections 2-b-1 and 2-b-2 of JPO79). Nonlinear boundary layers for both the ocean-ice and air-ice surface traction are used. A key component of this momentum balance is the force due to internal ice stress (eqs 5 and 6, JPO79). This force is based on a constitutive law which relates the ice stress to the strain rate and ice strength. For this model, a viscous-plastic constitutive law is used. Rigid plastic behavior is approximated in this law by allowing the ice to flow in a plastic manner for normal strain rates and to creep in a linear viscous manner for small strain rates (eq 4-11 and Fig. 1 and 2 in JPO79). The treatment of ice as a viscous plastic fluid was largely motivated by the desire to avoid the complexities associated with elastic plastic rheologies (e.g. Pritchard 1975) while still retaining plastic behavior under flow.

The second major component of the model consists of *continuity equations* (section 2-b-3, JPO79) describing the evolution of the ice thickness characteristics. These equations are essentially a simplification to two levels of the Thorndike et al. (1975) ice thickness destribution model. In this particular code two categories of ice are assumed : thin ice (<0.5 m in thickness) and thick ice (>0.5 m). To keep track of these two categories, two parameters are calculated: the mean ice thickness per unit area h and the compactness A which is defined as the fraction of area covered by thick ice. The continuity equations describing the evolution of these parameters (eq 13-16, JPO79) also include thermodynamic terms. For testing the model geographically invariant growth rates as a function of thickness and time of year were used.

A final component is an *ice strength*. For this two-level case the strength is taken to depend linearly upon thickness, and exponentially upon compactness (eq 17, JPO79).

Numerical scheme

The coupled nonlinear equations of the model are treated as an initial value problem using energy conserving finite difference techniques. A fixed grid is used so that the set of equations may be integrated as long


Atmospheric Pressure
Dynamical Input Fields-Wind and Current
Ocean Dynamic Height
Momentum Equations
$\underset{\sim}{C} + \underset{\sim}{\tau}_W + \underset{\sim}{\tau}_A + \underset{\sim}{F} + \underset{\sim}{G} + \underset{\sim}{T} = m \frac{D\underset{\sim}{u}}{Dt}$
Numerical Finite Difference Solution to Obtain Ice Velocities
Viscous Plastic Sea Ice Constitutive Law
Estimation of Ice Strength from Ice Thickness and Concentration
Dynamic Thermodynamic Evolution of Ice Thickness Characteristics (Ice mass and concentration)
Thermodynamic Model

C	*Coriolis Force*
τ_W	Water stress due to ice motion
τ_A	Air stress
F	Internal ice stress variation
G	Ocean currents
T	Ocean tilt
$\frac{D\underset{\sim}{u}}{Dt}$	Ice acceleration and momentum advection
m	Ice mass per unit area

Figure 1. Flow chart for two-level dynamic thermodynamic sea ice model.

as desired. Under this initial value approach, values of all parameters at all grid points are required to start the integration, and boundary values of velocity are required thereafter. The basic forcing fields for the model consist of geostrophic wind fields and vertically integrated ocean currents. Both inertial and momentum advection terms are retained in the dynamical equations. Because of the very strong ice interaction term, explicit integration of the momentum equations would force time steps to be extremely small (of the order of a few minutes). The ice thickness equations, on the other hand, can be explicitly integrated over time steps of several days. To avoid this severe time step limitation due to the ice interaction, the momentum equations are integrated implicitly. This implicit integration does, however, require a relaxation solution of a set of simultaneous equations at each time step. (Details on the relaxation scheme are given in App. A, JPO79.)

Using this implicit scheme the inertial terms will normally be significant for short time steps (of the order of 1 hour or less) but insignificant for long time steps. Although not a formal stability requirement, it is wise to choose time steps that are small compared to the variability of the ice forcing because the nonlinear terms are treated in a semi-implicit manner in the momentum balance. The effect of different time step magnitudes on the momentum balance is discussed in Appendix B of JPO79. This appendix also contains some examples which may help the user decide on time step magnitudes for particular applications. Because of this implicit treatment of the momentum equation, the only formal stability requirement is a Courant-Friedrichs-Lewy condition for the advection terms in the thickness equations:

$$\Delta t < \Delta x\,[2(u^2+v^2)]^{-1/2}.$$

A key feature of the numerical scheme is a staggered grid (Fig. 5, JPO79). This allows both the ice strength (i.e. the nonlinear viscosities) and the ice velocities to vary in space. Also this configuration greatly aids in constructing energy-conserving finite differences. The spatial finite differences used in the numerical scheme are described in some detail in Appendix A of JPO79. To a large degree this staggered grid is patterned after that employed in primitive equation numerical ocean models (e.g. Bryan 1969).

In the time-marching scheme (see eq 21-24, JPO79) ice thickness and velocity are temporally staggered. In particular, the thickness characteristics are considered to be defined at a temporal location halfway between the ice velocities. This scheme (formally called the forward-backward procedure) efficiently integrates coupled equations of this kind (e.g. Mesinger and Arakawa 1976).

As mentioned above, initial conditions at all points and ice velocities at the boundaries are thereafter required to initiate the integration of the system of equations forward in time. The most natural boundary condition is to take the ice velocity to be zero on the boundaries. This can be done either at a land boundary or at an ocean location where there is no ice. Note that the boundary condition does not affect the ice motion in such circumstances since in the absence of ice the strength is zero. More generally, as long as the ocean boundaries are removed from the ice edge, the coupled nature of the model will cause a natural ice edge boundary condition to be created (i.e. the ice strength drops to zero naturally near the edge). However, it is also possible to form an "open" boundary condition by setting the strength equal to zero near a boundary. In JPO79 this type of open boundary condition is used at the Spitsbergen-Greenland passage to allow natural inflow or outflow. In the computer code three boundary meshes are used to define arbitrary boundaries and "open" boundaries. Islands are also allowed. Consequently by simply modifying these meshes highly irregular boundaries may be taken into account.

Because of the strong ice interaction (which in this model is dissipative in nature) the momentum equations are essentially parabolic in form and hence have few numerical instability problems over long-term integrations. (While there are few such numerical problems it should be emphasized that the dissipative ice interaction terms are highly nonlinear and can lead to unstable flow fields in the absence of water drag. Such a feature is a physical characteristic of plastic flow and not a numerical artifact). However, in principle it is possible for the ice interaction to be very small even though there may be a finite ice mass. Under this situation, the momentum advection terms could cause nonlinear instabilities. To ensure against such situations, the bulk viscosity parameter (and hence indirectly shear viscosity as well) is never allowed to drop below 4.0×10^8 kg s^{-1}, a value which negligibly modifies the ice drift.

Nonlinear instabilities over long-term integrations can also arise from the nonlinear advection terms in the thickness continuity equations. To avoid this problem, small biharmonic and harmonic diffusion terms have been added to the continuity equation. These terms are grid-size dependent in the code and are automatically made smaller for finer mesh grids. These diffusion terms are treated in a mass-conserving manner at all boundaries (see App. A, JPO79). In practice these diffusive terms make only small contributions (in the JPO79 seasonal simulation the average diffusive flux was less than 3% of the average advective flux). It also appears that in the fully coupled model the diffusion terms might be dispensed with. Some sensitivity tests of the effects of diffusion are discussed in Appendix C of JPO79.

To integrate this model over very long time intervals requires a thermodynamic code specifying ice growth rates as a function of thickness and time of year. In the present code, spatially invariant growth rates (applicable to the Arctic Basin) from Thorndike et al. (1975) are used. For interested readers, time-independent computational procedures particularly useful for such growth calculations have been developed by Semtner (1976) for perennial multi-year ice, and Maykut (1978) for thin first-year ice.

Table 1. Schematic listing of model subroutines.

Level 1	*Level 2*	*Level 3*
	ADVECT (100-930)	DIFFUS (1190-1490)
	BNDRY (940-1180)	
	RELAX (8470-10720)	FELLD (1500-1730)
		FELLD1 (1740-1990)
		FELLIP (2000-2330)
		XMAXM (12120-12230)
MAIN (4880-7810)	FORM (2500-3540)	PLAST (8000-8460)
	GROWTH (4100-4870)	GROATE (3810-4090)
	GEO (3550-3800)	
	MEAN (7820-7990)	
	FGROWP (2340-2490)	
	XSUM (12240-12350)	

COMPUTER CODE

A complete test listing of the computer code is given in Appendix A. In this test code an idealized 5x6 grid is used. All arrays in this program are indexed beginning at the lower left-hand corner point (1, 1). Columns are denoted by constant x values, and rows by constant y values [i.e. the bottom row of an array consists of the points (1, 1), (2, 1) etc.]. In Appendix B double precision test run results of the code are presented.

In discussions of the code below, line numbers of specific statements will be given in parentheses. It is also noted that, while in principle the code is applicable to rectangular grid cells, it has only been tested for square grid cells ($D_x = D_y$).

Overall structure

The code is essentially divided into three levels. The first level consists of the main driving program (4880-7810) which sets up the boundaries, calls subroutines to numerically integrate the system of equations forward in time, and analyzes and stores results. (A description of the flow in the main program is given below.) The second level consists of various subroutines to numerically advance equations in time and to define various nonlinear parameters. These subroutines, in turn, call specialized subroutines at level 3 to, for example, estimate growth rates or determine nonlinear viscosities for plastic flow on a prescribed yield surface. Table 1 gives the schematic location; Table 2 briefly summarizes the function of each subroutine and Table 3 gives a brief description of the major variables. In all subroutines and the main driving program, a parameter statement is used to define dimension variables. Thus, by a simple editing command, array sizes can be modified everywhere in the code.

Note from the program listing in Appendix A that the thickness and compactness grids are one unit larger in dimension in each direction than the velocity grid. With the staggered grid configuration this allows thickness values to be defined outside the velocity boundaries, a procedure which is convenient for second order differencing in the thickness equations. As mentioned earlier key features of the code are thickness and velocity masks which define the boundaries. These are defined by subroutine BNDRY. Specifically, UVM is a mask for the ice velocity (1 at calculated velocity locations, 0 at boundaries; islands are allowed), HEFFM is the thickness mask (1 inside boundaries, 0 outside), and OUT is an outflow or open boundary mask (same as HEFFM except that it is zero at grid cells where an open boundary is required). For the test case the masks (defined in a file called BNDATA—see line 1005, Appendix A) create the grid configuration shown in Figure 2. At outflow grid cells the ice strength is set equal to zero and the ice mass removed. (All the mass removed is recorded in the main program.) Also, to approximately preserve second-order accuracy for advection into or out of an open cell, the mass and compactness in outflow grid cells are estimated by taking an average from all the grid cells adjacent to the open boundary (subroutine MEAN).

The input fields to the model consist of Geostrophic wind, Geostrophic Ocean currents (from which tilt is also calculated by the code) and growth rates. Other parameters are simulated (except for initial conditions and boundary values). Some description of the preparation of input fields from atmospheric pressure data and ocean currents is given in JPO79, section 3.

Appendix B gives the results of a 21-day test run of the complete code. For documentation purposes these test results are in double precision. In practice, however, double precision is not needed [except possibly, in the relaxation subroutine, subroutine RELAX, for very small grid sizes (say <25 km)].

Table 2. Brief description of model subroutines.

Name and Location	*Function*
MAIN (4880-7810)	Main driving program for model.
ADVECT (100-930)	Performs advection by explicit time stepping (modified Euler).
DIFFUS (1190-1490)	Performs diffusion by explicit forward time steps.
BNDRY (940-1180)	Sets up boundary masks (UVM, HEFFM, OUT).
RELAX (8470-10720)	Solves linearized momentum balance with spatially varying bulk and shear viscosities. Uses over-relaxation techniques.
FELLD (1500-1730) FELLD1 (1740-1990) FELLIP (2000-2330)	Calculate various finite differences for use in subroutine RELAX.
XMAXM (12120-12230)	Finds location of maximum value of a vector.
FORM (2500-3540)	Sets up forces, nonlinear water drag coefficients and nonlinear viscosities (for plastic flow) for use in each time step.
PLAST (8000-8460)	Calculates nonlinear viscosities based on plastic flow (specified by an elliptical yield curve) for subroutine FORM.
GROWTH (4100-4870)	Calculates changes in compactness A and mean ice thickness h due to growth, and redistribution by ridging.
GROATE (3810-4090)	Calculates growth rates for thin and thick ice in each grid cell.
GEO (3550-3800)	Reads in geostrophic wind data at each time step and ocean currents at the beginning of the model run.
MEAN (7820-7990)	Estimates thickness and compactness at "open boundary" grid cells based on adjacent values.
FGROWP (2340-2490)	Defines growth rates vs thickness. (In seasonal runs, this subroutine is not used. Instead a year-long matrix of growth rates is read in at line 5680 and daily values taken from this matrix at lines 6200-6230.)
XSUM (12240-12350)	Sums up values of a vector.

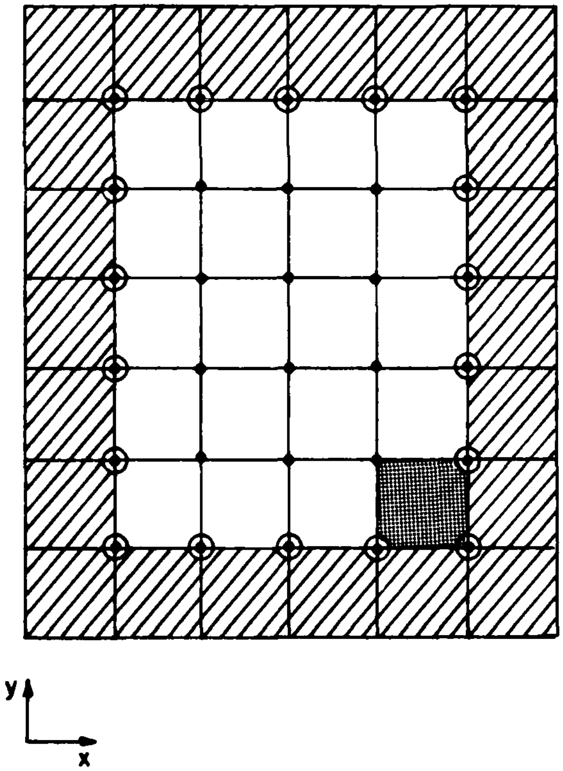


Figure 2. Boundary configuration for test grid. Circled grid points have zero velocities; shaded grid cells have zero thickness and compactness. Speckled grid cell is an "outflow" cell.

Description of flow in main driving program

To aid in understanding and using this code it is helpful to trace through the initialization and time-stepping procedure in the main driving program (lines 4880-7810). Such a procedure also gives some insight into various subtleties of the numerics as well as of the computer code. In this flow description, statements will be referred to by line number.

The first part of the main driving program (4880-6170) is concerned with initialization. Early in the code (5100-5220) constants are defined (1-day time steps, 125-km square grid, etc.) and various counters initialized (5230-5300). At lines 5310-5520 the boundary masks are read in (using subroutine BNDRY) and various sums over the boundaries performed to determine the number of grid cells covered by ice, etc. The program also prints out all the boundary masks to ensure that they are, in fact, the desired boundaries. At lines 5530-5640 the geostrophic currents are read in along with the first day's geostrophic wind data (using subroutine GEO). Note that a parameter WK (see line 5540) is used to cause GEO to either read in new currents or to use old ones. At line 5650-5690, a set of growth rates is read in (using subroutine FGROWP). In the main seasonal simulation in JPO79, FGROWP is ignored and a seasonal matrix is read off the tape at line 5680. However, for ice forecasting where the growth rates do not change very much, FGROWP with a data statement could have been used. FGROWP currently has January growth rates from Thorndike et al. (1975). Growth rates are entered

Table 3. Description of major variables (symbol in parentheses refers to variable symbol used in JPO79; e.g. h refers to ice thickness, A to compactness, etc.).

Symbol	*Description*	*Location in staggered grid (see Fig. 5, JPO79)*
	Arrays	
UICE (u)	x component of ice velocity	corner
VICE (v)	y component of ice velocity	corner
HEFF (h)	mean ice thickness per unit area	center
AREA (A)	compactness (fraction of area covered by thick ice)	center
ETA (η)	nonlinear shear viscosity	center
ZETA (ζ)	nonlinear bulk viscosity	center
AMASS (m)	ice mass per unit area	corner
DRAGS (D_s)	symmetric nonlinear water drag component parameter plus inertial term	corner
DRAGA (D_a+λ)	antisymmetric nonlinear water drag plus Coriolis parameter	corner
GAIRX [$\underset{\sim}{U}_g$ (x component)]	x component of geostrophic wind	corner
GAIRY [$\underset{\sim}{U}_g$ (y component)]	y component of geostrophic wind	corner
GWATX [$\underset{\sim}{U}_w$ (x component)]	x component of geostrophic ocean current	corner
GWATY [$\underset{\sim}{U}_w$ (y component)]	y component of geostrophic ocean current	corner
FORCEX [$\tau_x + (\partial P/\partial x)$]	x component of external force plus ice pressure gradient	corner
FORCEY [$\tau_y + (\partial P/\partial y)$]	y component of external forces plus ice pressure gradient	corner
UVM	boundary mask for velocity field	corner
HEFFM	boundary mask for ice thickness field	center
OUT	boundary mask for thickness excluding "open boundaries"	center
HCORR	amount of ice needed to remove "negative ice thickness"	center
UICEC VICEC }	intermediate ice velocities for use in semi-implicit time step of momentum balance	corner
OUT1	difference between HEFFM and OUT	center
OUT2	dummy variable for use in HMEAN	center
UERR VERR }	used to obtain velocity differences between time steps	corner
HDIFF	variable to keep track of diffusion	center
FHEFF [$f(h/A)$]	growth rate of thick ice	center
FO [f(o)]	growth rate of thin ice	center
STRESS (σ)	xx, yy, and xy components of ice stress (mks units)	center
HDF	variable to aid in keeping track of diffusion	center
COR (λ)	Coriolis parameter	corner
DAIRN ($=\rho_a C_a \vert U_g \vert$)	nonlinear air stress coefficient	corner
DWATN ($=\rho_w C_w \vert U_w - u \vert$)	nonlinear water stress coefficient	corner
PRESS (P)	ice pressure or strength	center
ZMAX (ζ_{max})	maximum allowable bulk viscosity value	center
ZMIN (ζ_{min})	minimum allowable bulk viscosity value	center
GHEFF	average growth tendency	center
GAREA	area change tendency	center
E11 (ϵ_{xx})	xx strain rate component	center
E22 (ϵ_{yy})	yy strain rate component	center
E12 (ϵ_{xy})	xy strain rate component	center
	Vectors	
FGROW (f)	growth rate of ice at 0.5-m interval [There are also a variety of vectors in the program which are equivalenced to arrays. This was done to facilitate sums and maximum point location which can be vectorized when vectors are used on the Texas Instruments Advanced Scientific Computer (ASC) for which this code was initially written.]	center

Table 3 (cont'd). Description of major variables (symbol in parentheses refers to variable symbol used in JP079, e.g. h refers to ice thickness, A to compactness, etc.).

Symbol	*Description*
	Constants
DELTT AND DELTAT	time step length in seconds
DELTAX	grid size in meters
DELTAY	grid size in meters
ERROR	accuracy of relaxation solution in m/s.
DIFF1 (D_1)	harmonic diffusion constant
A22	minimum allowable value of compactness
HO (h_0)	demarcation thickness between thin and thick ice
RHOICE (ρ_I)	ice density
FCOR (λ)	average Coriolis parameter
RHOAIR (ρ_a)	air density
SINWIN	sine of air turning angle
COSWIN	cosine of air turning angle
SINWAT	sine of ocean turning angle
COSWAT	cosine of ocean turning angle
GRAV (g)	acceleration of gravity
ECCEN (e)	ratio of principal axes of plastic yield ellipse
PSTAR (P^*)	strength constant
THETA	weighting for momentum time stepping. THETA = 1.0 yields a backwards time step; THETA = 0.5 a Crank-Nicholsen centered time step and 0.0 a forward explicit time step.

in units of centimeters per day and converted to meters of ice per second. Lines 5700-5930 initialize the ice thickness and velocity. [Note that the velocity, thickness and compactness functions are defined at three time levels, e.g. H (I, J, K) K = 1, 2, 3. Of these values the K = 1 level is always the present value and the K = 2 level the previous time step.] As part of this initialization, line 5810 calls a subroutine to sum up the thicknesses at level 1 over the grid, excluding the "outflow" grid cells. This is accomplished by using the vector HEFF1 which is equivalenced to the thickness array HEFF. (Note that the length of HEFF1 is such that only the first level of HEFF is included in HEFF1. This procedure has been followed on almost all the equivalence vectors.) In this initialization, the ice is set to 100% compactness with a uniform thickness of 3.2967m (≐ 3.0/density of ice).

The remainder of the initialization (5940-6160) is concerned with estimating the initial ice velocity field. (For long-term integrations this step is not critical.) To do this a spatially constant velocity is chosen, the ice mass is set equal to zero (effectively removing the inertial term from the relaxation solution), and the parameter THETA is set equal to 1. This procedure gives a backwards time step–see eqs 21 and 22, JPO79–so that the relaxation solution solves for the entire ice velocity rather than portions of it, as would happen in a centered time step, e.g. THETA = 0.5.

Having initialized the variables, the standard time-stepping procedure is begun at line 6170. After the conclusion of a given time step, the program returns to line 6180 (more precisely to FORTRAN statement number 100) and cycles through again. To initiate the integration, growth rates applicable to the day under consideration are defined from the growth rate matrix (line 6220). Since the present wind field is still valid, a new wind field is not called in until the end of the time step–to be mentioned later. Since only fixed January growth rates are used in the test case, line 6220 is commented. (Note that in this particular FORTRAN program comments are denoted by asterisks.) In lines 6240-6351 thicknesses and compactnesses at outflow points are estimated using subroutine MEAN. Basically this subroutine uses grid cells containing ice adjacent to the "open" boundaries to estimate ice thicknesses for use in advection. All the ice flowing into the "open" boundary cell is, however, explicitly accounted for. Part of this accounting is done at line 6351 where the amount of ice in the open cells (THEFF1) at the beginning of the time step is calculated.

At lines 6360-6550 the "predictor" portion of the momentum time step is performed (see eq 21, JPO79). A feature of this process is that at lines 6390-6420 ice velocities at the third level [e.g. UICE (I, J, 3)] and "centered" ice velocities [UICEC (I, J)] are set equal to the present ice velocities (level one) before calling

RELAX. This is done for two reasons: 1) the initial velocity "guess" to start the relaxation procedure is the ice velocity at the third level and 2) the momentum advection term is linearized by using UICEC (see eq 21 and App. A, JPO79). To carry out the predictor procedure, the time step is halved (line 6510) and FORM is called. FORM uses the present ice velocity to linearize the momentum equations as discussed earlier.

In lines 6560-6760 the main forward time step for the momentum balance is carried out (see eq 22, JPO79). The procedure here is the same as in the predictor step except that a full time step is used and the "predicted" velocity is employed to estimate the various nonlinear terms. This predictor corrector procedure approximately centers the nonlinear terms while still maintaining a backward time step for the "diffusion" terms.

Following the momentum time step, the thickness, growth and decay terms are explicitly stepped forward in time in lines 6770-6820. These operations correspond to the finite difference equations (eq 23-24) in JPO79. The advection subroutine (ADVECT) handles the dynamical portions of the compactness and thickness continuity equations, while the growth subroutine treats the thermodynamic terms. (The growth subroutine would be the logical place to insert a more complete heat budget subroutine.) As noted at line 6820, it is important to call the growth subroutine after the advection subroutine. This is because GROWTH also handles the mechanical redistribution by insisting that A be less than or equal to 1. GROWTH also corrects for problems such as negative ice thicknesses which may arise because centered differencing of second-order accuracy is used in the advection code (such negative values can be avoided by upstream differencing, but such schemes are excessively diffusive). Numerical studies of advection (Mahlman and Sinclair 1977) show that such negative value removal substantially aids in tracer accuracy provided that it is applied immediately as done here. Note that in these thickness time steps the latest values of the ice velocity are used. As mentioned earlier, this effectively means that the coupled thickness-momentum equations are staggered in time (see eq 21-24, JPO79).

Once the time stepping is completed, much of the remainder of the code is devoted to checking various sums over the array to ensure conservation and to keep track of the various contributions to the thickness changes. In lines 6830-6933, the outflow ice and various average thicknesses and growth sums are calculated. From 6940-7230 complete information over the grid (at an interval specified by lines 6950-6955) is printed out. In lines 7240-7420 velocity change information is printed out (over the same interval). In lines 7440-7510 simulated data for each time step are written to tape for later analysis (these write statements are given comments in the test code). From lines 7511-7516 sum information is printed out (every time step). In lines 7520-7640 new winds are read in, and the wind data file is rewound if a whole year is over. Finally a check is made on the main time-step counter to see if the integration is finished. If not, the program cycles back to FORTRAN line 100. If finished, the program terminates after writing out some restart parameters (line 7790) for continuing the integrations. Some of the other termination steps after line 7650 are at present superfluous. They were initially inserted for an examination of "spin down" characteristics of the model.

Some comments on the relaxation subroutine

Since the relaxation code is the heart of the dynamic equations, some brief comments on this subroutine are added here. As presently written, the relaxation subroutine is unnecessarily complex. There are several reasons for this, mostly based on computational efficiency considerations. For one, the relaxation code was separately converted to single precision for the seasonal runs discussed in JPO79 to speed up the code. For this purpose all variables except those beginning in O were defined in single precision. As a result, the input double precision variables are converted to single precision at the beginning of the code by a variety of DO loops. Since such loops are vectorizable they take negligible time on the ASC. In addition the subroutines FELLD, FELLD1 and FELLIP were converted to single precision. While some changes may well be useful, caution should be exercised in any tinkering here. This is because double-precision and single precision variables are sometimes necessarily mixed in the code due to certain ASC compiler peculiarities.

A second reason for the complexity of the relaxation code, also related to computational efficiency, is that a large portion of the relaxation sweep was vectorized by calculating separately (in the subroutine FELLD1) as much of the finite difference code as can be performed over the whole array without replacement. While increasing the complexity of the code, this "separation" of the finite differences more than doubles the speed of the relaxation sweep on the ASC.

A third point is that the relaxation code was made general with respect to backward or forward time steps. In particular by adjusting THETA (input variable OTHETA) backwards, centered (Crank-Nicholson) or forward time steps may be used. To effect this generality, stresses at the previous time step are constructed by calling subroutine FELLIP. In general the backwards time step (THETA = 1.0) is most stable but as mentioned in JPO79 it does overdamp inertial waves. In the Crank-Nicholson scheme (THETA = 0.5, see, for example, Richtmyer and Morton 1967) this does not occur. Consequently such a scheme is probably preferable at smaller time steps, say < 1 hour. At longer time steps, however, splitting problems have been found to occur with the centered scheme. These problems arise because the inertial term is the only term pulling alternating velocity variations together and this term is usually small at large time steps. Some brief discussion of this is given in section 2-C of JPO79.

A final point is that after 100 sweeps of over-relaxation, the subroutine returns to straight relaxation. This is because, while normally faster, the over-relaxation procedure can sometimes diverge.

CONCLUDING REMARKS

As presently configured, this sea ice model has direct application to long-term simulations. However, it also has many features which may well prove useful in ice forecasting. In addition, aspects of the code may aid in simulating plastic flow in geophysical media other than sea ice. In both such applications, modification of this code will probably be helpful. In these cases, features of the code either relevant or necessary for long integrations may be less important. In addition, certain characteristics which play no major role in long-term simulations may prove important in forecasting.

A feature important for the long-term studies but less important for forecasting is the presence of nonlinear (and to a lesser degree linear) instabilities. The main examples of these are alternating grid point instabilities occurring in the thickness and compactness fields due to the nonlinear advection terms. To a lesser extent similar problems may arise in the momentum equation due to the momentum advection terms. Physically these instabilities arise due to the nonlinear advection terms causing a cascade of energy to higher wave numbers. Due to the finiteness of the grid, this energy tends to pile up at the Nyquist wave number, producing high wave number instabilities. Because of the compressible nature of sea ice flow, conventional energy-conserving techniques cannot always prevent this. In this code, this energy cascade is largely removed by biharmonic diffusion. This diffusion preferentially damps out the high wave number energy while little affecting the lower wave number. (In some sense this procedure can be thought of as a "poor man's spectral model.") In short-term integrations, however, such instabilities never have time to build up. Consequently, in such applications there appears to be no problem in removing all diffusion terms in the thickness equations. This removal will also cause some slight linear instability (since the advection terms are not perfectly centered in time). However, over short time intervals these instabilities are minor. For similar reasons these considerations also apply to the minimum ice viscosity used in the momentum equations.

A converse feature, which is more critical in the short term, is the problem of initialization. For long-term simulations, the model can be integrated long enough so that it eventually "forgets" the initial conditions. For short-term forecasts, however, the initial conditions largely dominate the forecast. How to minimize the impact of poor initial conditions is an area needing considerable research. One approach may be to use observed forcing data to run the model for a few days prior to the forecast. Thus the model may generate some of its own initial conditions. It may also be that update data can be continually assimilated together with this "initialization" run.

Another difficulty results from specifying boundary conditions for localized regions. Persistence may be useful here. In any case a variety of these problems need numerical examination.

Finally, features of perhaps more general interest are some of the plastic characteristics of sea ice flow. In ice models to date many of these characteristics have been suppressed due to the damping nature of the oceanic boundary layer. In the absence of this damping, however, the plastic flow is inherently unstable and becomes ill-defined without inclusion of the inertial terms. In particular, in the absence ot water drag, the system will begin to accelerate once the plastic yield is exceeded. This behavior has been verified directly with this code. (Initial results suggest, however, that this particular phenomenon is not critically dependent on time-step magnitude.) Consideration of spatially varying ocean coupling effects with the attendant buildup of pressure terms may well further complicate these plastic flow fields. Also on shorter time scales, inertial oscillation in both the ice and ocean may be important. Investigation of such effects is needed. In these investigations, aspects of the code described here may prove useful.

LITERATURE CITED

Bryan, K. (1969) A numerical method for the study of the circulation of the world oceans. *Journal of Computational Physics*, vol. 4, p. 347-376.

Hibler, W.D., III (1979) A dynamic thermodynamic sea ice model. *Journal of Physical Oceanography*, vol. 9, no. 4, p. 815-846.

Mahlman, J.D. and R.W. Sinclair (1977) Tests of various numerical algorithms applied to a simple trace constituent air transport problem. *Fate of Pollutants in the Air and Water Environments*, Part 2, vol. 8 (I.H. Suffet, ed.), New York: Wiley, p. 223-252.

Maykut, G.A. (1978) Energy exchange over young sea ice in the central arctic. *Journal of Geophysical Research*, vol. 83, p. 3646-3658.

Mesinger, F. and A. Arakawa (1976) *Numerical methods used in atmospheric models.* GARP Publication, no. 17, Geneva.

Pritchard, R.S. (1975) An elastic-plastic constitutive law for sea ice. *Journal of Applied Mechanics*, vol. 42E, p. 379-384.

Richtmyer, R.D. and K.W. Morton (1967) *Difference methods for initial-value problems.* New York: Wiley.

Semtner, A.J., Jr. (1976) A model for the thermodynamic growth of sea ice in numerical investigations of climate. *Journal of Physical Oceanography*, vol. 6, p. 379-389.

Thorndike, A.S., D.A. Rothrock, G.A. Maykut and R. Colony (1975) The thickness distribution of sea ice. *Journal of Geophysical Research*, vol. 80, no. 33, p. 4501-4513.

APPENDIX A: LISTING OF COMPUTER CODE OF TWO–LEVEL MODEL.

```
      SUBROUTINE ADVECT(UICE1,VICE1,HEFF,DIFF1,LAD)
* SPACER
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1
      DIMENSION HEFF(NX1,NY1,3),UICE1(NX,NY,3),VICE1(NX,NY,3)
     &,UICE(NX,NY),VICE(NX,NY)
      COMMON/STEP/DELTAT,DELTAX,DELTAY
      COMMON/ARRAY/HEFFM(NX1,NY1),UVM(NX,NY)
      COMMON/DIFF/HDIFF(NX1,NY1)
* NOW DECIDE IF BACKWARD EULER OR LEAPFROG
      LL=LAD
      IF(LL EQ 1) GO TO 100
* BACKWARD EULER
      DELTT=DELTAT
      K3=2
      K2=2
      GO TO 101
* LEAPFROG
 100  DELTT=DELTAT*2.0
      K3=3
      K2=2
 101  CONTINUE
* NOW REARRANGE H'S
      DO 401 J=1,NY
      DO 401 I=1,NX
      UICE(I,J)=UICE1(I,J,1)
      VICE(I,J)=VICE1(I,J,1)
 401  CONTINUE
      DO 200 J=1,NY1
      DO 200 I=1,NX1
      HEFF(I,J,3)=HEFF(I,J,2)
      HEFF(I,J,2)=HEFF(I,J,1)
      HDIFF(I,J)=0.0
 200  CONTINUE
 202  CONTINUE
* NOW GO THROUGH STANDARD CONSERVATIVE ADVECTION
      DELTX=DELTT/(4.0*DELTAX)
      DELTY=DELTT/(4.0*DELTAY)
      DO 210 J=1,NYM1
      DO 210 I=1,NXM1
      HEFF(I+1,J+1,1)=HEFF(I+1,J+1,K3)-DELTX*((HEFF(I+1,J+1,2)+HEFF
     &(I+2,J+1,2))*(UICE(I+1,J+1)+UICE(I+1,J))-(HEFF(I+1,J+1,2)+HEFF
     &(I,J+1,2))*(UICE(I,J+1)+UICE(I,J)))-DELTY*((HEFF(I+1,J+1,2)+HEFF
     &(I+1,J+2,2))*(VICE(I,J+1)+VICE(I+1,J+1))-(HEFF(I+1,J+1,2)+HEFF
     &(I+1,J,2))*(VICE(I,J)+VICE(I+1,J)))
 210  CONTINUE
* NOW DECIDE IF DONE
      IF(LL.EQ.2) GO TO 99
      IF (LL.EQ.3) GO TO 89
      GO TO 102
 89   CONTINUE
*  NOW FIX UP H(I,J,2)
      DO 88 J=1,NY1
      DO 88 I=1,NX1
      HEFF(I,J,2)=HEFF(I,J,3)
 88   CONTINUE
      GO TO 102
 99   CONTINUE
* NOW DO BACKWARD EULER CORRECTION
      DO 220 J=1,NY1
      DO 220 I=1,NX1
      HEFF(I,J,3)=HEFF(I,J,2)
      HEFF(I,J,2)=0.5*(HEFF(I,J,1)+HEFF(I,J,2))
 220  CONTINUE
      LL=3
```

```
740        K3=3
750        GO TO 202
760   102  CONTINUE
770 * NOW DO DIFFUSION ON H(I,J,K3)
780        DO 240 KD=1,2
790        GO TO (241,242),KD
800   241  CONTINUE
810        CALL DIFFUS(UICE,VICE,HEFF,DIFF1,DELTT)
820        GO TO 243
830   242  CONTINUE
840        DIFF2=-(DELTAX**2)/DELTT
850        CALL DIFFUS(UICE,VICE,HEFF,DIFF2,DELTT)
860   243  CONTINUE
870        DO 330 J=1,NY1
880        DO 330 I=1,NX1
890        HEFF(I,J,1)=(HEFF(I,J,1)+HEFF(I,J,3))*HEFFM(I,J)
895        HDIFF(I,J)=HDIFF(I,J)+HEFF(I,J,3)*HEFFM(I,J)
900   330  CONTINUE
910   240  CONTINUE
920        RETURN
930        END
940        SUBROUTINE BNDRY
950 * SUBROUTINE SETS UP BOUNDARY MASK
960        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
970        PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
980       &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1,N4=NX*NY
990        COMMON/ARRAY/HEFFM(NX1,NY1),UVM(NX,NY)
1000        COMMON/OUTFLO/OUT(NX1,NY1)
1005        OPEN(UNIT=1,FILE="BNDATA")
1010 * READ IN VELOCITY MASK
1020        READ(UNIT=1,FMT=9)((UVM(I,J),I=1,NX),J=1,NY)

1030  9     FORMAT(30G1.0)
1040        READ(UNIT=1,FMT=8)((HEFFM(I,J),I=1,NX1),J=1,NY1)
1050        READ(UNIT=1,FMT=8)((OUT(I,J),I=1,NX1),J=1,NY1)
1060  8     FORMAT(42G1.0)
1170        RETURN
1180        END
1190        SUBROUTINE DIFFUS(UICE,VICE,HEFF,DIFF1,DELTT)
1200 * SPACER
1210        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
1220        PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
1230       &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1
1240        DIMENSION HEFF(NX1,NY1,3),UICE(NX,NY),VICE(NX,NY)
1250       &,HEFF1(NX1,NY1)
1260        COMMON/STEP/DELTAT,DELTAX,DELTAY
1270        COMMON/ARRAY/HEFFM(NX1,NY1),UVM(NX,NY)
1280 * SUBROUTINE DIFFUSES HEFF,MULTIPLIES BY DELT, AND PUTS RESULTS IN HEFF
1290 *  NOW ZERO OUT HEFF1
1300        DO 210 J=1,NY1
1310        DO 210 I=1,NX1
1320        HEFF1(I,J)=0.0
1330  210   CONTINUE
1340 * NOW DO DIFFUSION
1350        DELTXX=DELTT*DIFF1/(DELTAX**2)
1360        DELTYY=DELTT*DIFF1/(DELTAY**2)
1370        DO 220 J=2,NY
1380        DO 220 I=2,NX
1390        HEFF1(I,J)=DELTXX*((HEFF(I+1,J,3)-HEFF(I,J,3))*HEFFM(I+1,J)
1400       &    -(HEFF(I,J,3)-HEFF(I-1,J,3))*HEFFM(I-1,J))
1410       &  +DELTYY*((HEFF(I,J+1,3)-HEFF(I,J,3))*HEFFM(I,J+1)
1420       &  -(HEFF(I,J,3)-HEFF(I,J-1,3))*HEFFM(I,J-1))
1430  220   CONTINUE
1440        DO 260 J=1,NY1
1450        DO 260 I=1,NX1
1460        HEFF(I,J,3)=HEFF1(I,J)
```

```
 260  CONTINUE
      RETURN
      END
      SUBROUTINE FGROWP
* SUBROUTINE TO SET UP CANONICAL GROWTH VALUES
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1
      DIMENSION FGGOW(21)
      COMMON/GROWTH/FGROW(21)
      DATA FGGOW/12.09D+00,1.95D+00,.46D+00,.37D+00,.31D+00,.27D+00
     &,.21D+00,.14D+00,.09D+00,.03D+00,0.0,-.02D+00,-.03D+00,-.04D+00
     &,-.05D+00,-.05D+00,-.06D+00,-.06D+00,-.06D+00,-.06D+00,-.06D+00/
      DO 101 I=1,21
      FGROW(I)=FGGOW(I)/(8.64D+06)
      PRINT 1,I,FGROW(I)
 101  CONTINUE
 1    FORMAT(1X,I5,G20.12)
      RETURN
      END
      SUBROUTINE FORM(UICE,VICE,ETA,ZETA,DRAGS
     &,DRAGA,GAIRX,GAIRY,GWATX,GWATY,FORCEX,FORCEY,HEFF,AMASS,AREA)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1,N4=NX*NY
* PROGRAM FORMS BASIC INPUT PARAMETERS FOR RELAXATION
      DIMENSION UICE(NX,NY,3),VICE(NX,NY,3),ETA(NX1,NY1)
     &,ZETA(NX1,NY1),DRAGS(NX,NY),DRAGA(NX,NY),GAIRX(NX,NY)
     &,GAIRY(NX,NY),GWATX(NX,NY),GWATY(NX,NY),HEFF(NX1,NY1,3)
     &,COR(NX,NY),FORCEX(NX,NY),FORCEY(NX,NY),AMASS(NX,NY)
     &,DAIRN(NX,NY),DWATN(NX,NY),AREA(NX1,NY1,3),PRESS(NX1,NY1)
     &,ZMAX(NX1,NY1),ZMIN(NX1,NY1)
      COMMON/STEP/DELTAT,DELTAX,DELTAY
      COMMON/ARRAY/HEFFM(NX1,NY1),UVM(NX,NY)
      COMMON/OUTFLO/OUT(NX1,NY1)
* FIRST SET UP BASIC CONSTANTS
      DWAT=0.59D+00
      DAIR=0.01462D+00
      RHOICE=0.91D+03
      FCOR=1.46D-04
      RHOAIR=1.3D+00
      SINWIN=0.4226D+00
      COSWIN=0.9063D+00
      SINWAT=0.4226D+00
      COSWAT=0.9063D+00
      GRAV=9.832D+00
      ECCEN=2.0
* 20 DEG GIVES SIN EQUAL TO .34202
* NOW SET UP MASS PER UNIT AREA AND CORIOLIS TERM
      DO 101 J=1,NY
      DO 101 I=1,NX
      AMASS(I,J)=RHOICE*0.25*(HEFF(I,J,1)
     &  +HEFF(I+1,J,1)+HEFF(I,J+1,1)+HEFF(I+1,J+1,1))
      COR(I,J)=AMASS(I,J)*FCOR
 101  CONTINUE
* NOW SET UP NON LINEAR WIND AND WATER DRAG
      DO 99 J=1,NY
      DO 99 I=1,NX
      DAIRN(I,J)=RHOAIR*.0012D+00*DSQRT(GAIRX(I,J)**2+GAIRY(I,J)**2)
      DWATN(I,J)=5.5D+00*DSQRT((UICE(I,J,1)-GWATX(I,J))**2
     &  +(VICE(I,J,1)-GWATY(I,J))**2)
 99   CONTINUE
* NOW SET UP SYMMETTRIC DRAG
      DO 102 J=1,NY
      DO 102 I=1,NX
      DRAGS(I,J)=DWATN(I,J)*COSWAT
 102  CONTINUE
* NOW SET UP ANTI SYMMETRIC DRAG PLUS CORIOLIS
      DO 103 J=1,NY
      DO 103 I=1,NX
      DRAGA(I,J)=DWATN(I,J)*SINWAT+COR(I,J)
 103  CONTINUE
* NOW SET UP FORCING FIELD
      DO 104 J=1,NY
      DO 104 I=1,NX
* FIRST DO WIND
      FORCEX(I,J)=DAIRN(I,J)*(COSWIN*GAIRX(I,J)
     &-SINWIN*GAIRY(I,J))
      FORCEY(I,J)=DAIRN(I,J)*(SINWIN*GAIRX(I,J)
     &  +COSWIN*GAIRY(I,J))
* NOW ADD IN CURRENT FORCE
      FORCEX(I,J)=FORCEX(I,J)+DWATN(I,J)*(COSWAT*GWATX(I,J)
     &     -SINWAT*GWATY(I,J))
      FORCEY(I,J)=FORCEY(I,J)+DWATN(I,J)*(SINWAT*GWATX(I,J)
     &    +COSWAT*GWATY(I,J))
* NOW ADD IN TILT
      FORCEX(I,J)=FORCEX(I,J)-COR(I,J)*GWATY(I,J)
      FORCEY(I,J)=FORCEY(I,J)+COR(I,J)*GWATX(I,J)
 104  CONTINUE
* NOW SET UP ICE PRESSURE AND VISCOSITIES
* FIRST SET UP CONSTANTS
      PSTAR=5.0D+03
      EPSIL=5.787D-08
*     5.787D-08 IS 0.5 PER CENT PER DAY STRAIN RATE
      ZETAC=0.15
      ETAC=ZETAC/4.0
* NOW SET UP VALUES
      DO 105 J=1,NY1
      DO 105 I=1,NX1
      PRESS(I,J)=PSTAR*HEFF(I,J,1)*DEXP(-20.0*(1.0-AREA(I,J,1)))
      ZMAX(I,J)=(5.0D+12/(2.0D+04))*PRESS(I,J)
      ZMIN(I,J)=4.0D+08
*     ZETA(I,J)=ZETAC*PRESS(I,J)/EPSIL
*     ETA(I,J)=ETAC*PRESS(I,J)/EPSIL
*     ZETA(I,J)=DMAX1(4.0D+08,ZETA(I,J))
*     ETA(I,J)=DMAX1(1.0D+08,ETA(I,J))
 105  CONTINUE
      CALL PLAST(UICE,VICE,PRESS,ETA,ZETA,ECCEN,ZMAX,ZMIN)
* NOW SET VISCOSITIES AND PRESSURE EQUAL TO ZERO AT OUTFLOW PTS
      DO 106 J=1,NY1
      DO 106 I=1,NX1
      PRESS(I,J)=PRESS(I,J)*OUT(I,J)
      ETA(I,J)=ETA(I,J)*OUT(I,J)
      ZETA(I,J)=ZETA(I,J)*OUT(I,J)
 106  CONTINUE
* NOW CALCULATE PRESSURE FORCE AND ADD TO EXTERNAL FORCE
      DO 107 J=1,NY
      DO 107 I=1,NX
      FORCEX(I,J)=FORCEX(I,J)-(0.25/DELTAX)*
     &   (PRESS(I+1,J)+PRESS(I+1,J+1)-PRESS(I,J)-PRESS(I,J+1))
      FORCEY(I,J)=FORCEY(I,J)-(0.25/DELTAY)*
     &  (PRESS(I,J+1)+PRESS(I+1,J+1)-PRESS(I,J)-PRESS(I+1,J))
* NOW PUT IN MINIMAL MASS FOR TIME STEPPING CALCULATIONS
 107  CONTINUE
      RETURN
      END
      SUBROUTINE GEO(GAIRX,GAIRY,GWATX,GWATY,WK)
* IF WK LT 0 THEN READ IN CURRENTS,IF WK GT 100 THEN ZERO OUT CURRENTS
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1,N4=NX*NY
```

```
      DIMENSION GAIRX(NX,NY),GAIRY(NX,NY),GWATX(NX,NY),GWATY(NX,NY)
* FIRST READ IN WIND
      DO 101 J=1,NY
      DO 101 I=1,NX
      GAIRX(I,J)=3.0D+00+0.1D+00*(I+J)
      GAIRY(I,J)=-4.0D+00+0.2D+00*(I+J)
      GWATX(I,J)=0.01D+00-0.001D+00*(I-J)
      GWATY(I,J)=0.05D+00+0.001D+00*(I+J)
 101  CONTINUE
      RETURN
      END
      SUBROUTINE GROATE(H,HO,FH,FO,AREA)
* SUBROUTINE TO OBTAIN THICK AND THIN GROWTH RATES
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1
      DIMENSION H(NX1,NY1),FO(NX1,NY1),FH(NX1,NY1)
     &,AREA(NX1,NY1,3)
      COMMON/GROWTH/FGROW(21)
      ONE=1.0
      ZERO=0.0
      S2=HO+1.0
      M2=IDINT(S2)
      DO 101 J=1,NY1
      DO 101 I=1,NX1
      S1=H(I,J)/0.5+1.0
      S1=DMIN1(S1,2.0D+01)
      M1=IDINT(S1)
      FO(I,J)=(FGROW(M2+1)-FGROW(M2))*(S2-DFLOAT(M2))+FGROW(M2)
* THIN ICE IS ALWAYS TAKEN TO BE ZERO THICKNESS
 201  FO(I,J)=FGROW(1)
 202  CONTINUE
      FH(I,J)=(FGROW(M1+1)-FGROW(M1))*(S1-DFLOAT(M1))+FGROW(M1)
 101  CONTINUE
      DO 102 J=1,NY1
      DO 102 I=1,NX1
      FH(I,J)=FH(I,J)*AREA(I,J,2)+FO(I,J)*(1.0-AREA(I,J,2))
 102  CONTINUE
      RETURN
      END
      SUBROUTINE GROWTH(HEFF,AREA,FH,HO,A22)
* SUBROUTINE TO CALCULATE CHANGE OF PARAMETERS DUE TO GROWTH
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1
*  THIS SUBROUTINE MUST BE CALLED AFTER ADVECTION
* FH IS ADDITION MELTING THAT MUST BE DONE TO BALANCE MIXED LAYER
      DIMENSION HEFF(NX1,NY1,3),AREA(NX1,NY1,3),GHEFF(NX1,NY1)
     &,GAREA(NX1,NY1),H(NX1,NY1),FH(NX1,NY1)
      COMMON/RATE/FHEFF(NX1,NY1),FO(NX1,NY1)
      COMMON/STEP/DELTAT,DELTAX,DELTAY
      COMMON/ARRAY/HEFFM(NX1,NY1),UVM(NX,NY)
      COMMON/OUTFLO/OUT(NX1,NY1)
      ZERO=0.0
      ONE=1.0
* FIRST SOLVE FOR THICKNESS OF THICK ICE
      DO 101 J=1,NY1
      DO 101 I=1,NX1
      AREA(I,J,2)=DMAX1(A22,AREA(I,J,2))
      H(I,J)=HEFF(I,J,2)/AREA(I,J,2)
 101  CONTINUE
* GROWTH SUBROUTINE CALCULATES TOTAL GROWTH TENDENCIES INCLUDING SNOWFAL
      CALL GROATE(H,HO,FHEFF,FO,AREA)
 1    FORMAT(1X,11G11.5)
* NOW CALCULATE CORRECTED GROWTH
      DO 201 J=1,NY1
      DO 201 I=1,NX1
      GHEFF(I,J)=-DELTAT*FHEFF(I,J)
      GAREA(I,J)=DELTAT*FO(I,J)
 201  CONTINUE
      DO 102 J=1,NY1
      DO 102 I=1,NX1
      GHEFF(I,J)=-1.0*DMIN1(HEFF(I,J,1),GHEFF(I,J))
      GAREA(I,J)=DMAX1(ZERO,GAREA(I,J))
 102  CONTINUE
      DO 103 J=1,NY1
      DO 103 I=1,NX1
      FH(I,J)=DMIN1(ZERO,GHEFF(I,J))
 103  CONTINUE
* NOW DO AREA CHANGE
      DO 104 J=1,NY1
      DO 104 I=1,NX1
      GAREA(I,J)=2.0*(1.0-AREA(I,J,2))*GAREA(I,J)/HO
     &+.5*FH(I,J)*AREA(I,J,2)/(HEFF(I,J,1)+.00001)
 104  CONTINUE
* NOW CORRECT AREA AND HEFF(I,J,1)
      DO 105 J=1,NY1
      DO 105 I=1,NX1
      AREA(I,J,1)=AREA(I,J,1)+GAREA(I,J)
      HEFF(I,J,1)=HEFF(I,J,1)+GHEFF(I,J)*OUT(I,J)
 105  CONTINUE
* NOW ZERO OUTSIDE POINTS
      DO 109 J=1,NY1
      DO 109 I=1,NX1
      AREA(I,J,1)=AREA(I,J,1)*HEFFM(I,J)
      HEFF(I,J,1)=HEFF(I,J,1)*HEFFM(I,J)
 109  CONTINUE
* NOW SET AREA(I,J,1)=0 WHERE NO ICE IS
      DO 110 J=1,NY1
      DO 110 I=1,NX1
      AREA(I,J,1)=DMIN1(AREA(I,J,1),HEFF(I,J,1)/.0001)
 110  CONTINUE
* NOW TRUNCATE AREA
      DO 106 J=1,NY1
      DO 106 I=1,NX1
      AREA(I,J,1)=DMIN1(ONE,AREA(I,J,1))
 106  CONTINUE
      DO 107 J=1,NY1
      DO 107 I=1,NX1
      AREA(I,J,1)=DMAX1(A22,AREA(I,J,1))
 107  CONTINUE
* NOW CALCULATE ADDITIONAL ICE TO BE MELTED FOR MIXED LAYER BALANCE
      DO 108 J=1,NY1
      DO 108 I=1,NX1
      FH(I,J)=GHEFF(I,J)-DELTAT*FHEFF(I,J)
* NOW STORE COMMON GROWTH RATE
      FHEFF(I,J)=GHEFF(I,J)
 108  CONTINUE
      RETURN
      END
* MAIN DRIVING PROGRAM FOR VISCOUS PLASTIC SEA ICE MODEL
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
* SPACER
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1,N4=NX*NY
      DIMENSION UICE(NX,NY,3),VICE(NX,NY,3),ETA(NX1,NY1)
     &,ZETA(NX1,NY1),DRAGS(NX,NY),DRAGA(NX,NY),GAIRX(NX,NY)
     &,GAIRY(NX,NY),GWATX(NX,NY),GWATY(NX,NY),HEFF(NX1,NY1,3)
     &,AMASS(NX,NY),FORCEX(NX,NY),FORCEY(NX,NY),AREA(NX1,NY1,3)
     &,HCORR(NX1,NY1),HEFF1(N3)
     &,UICEC(NX,NY),VICEC(NX,NY)
     &,OUT1(NX1,NY1),OUT2(NX1,NY1),UERR(NX,NY),VERR(NX,NY),UERRV(N4)
```

```
5000       &,VERRV(N4)
5010       &,FGRW(365,21),HDIFF(NX1,NY1),FHE1(N3),GR(N3),AREA1(N3)
5020        COMMON/RATE/FHEFF(NX1,NY1),FO(NX1,NY1)
5030        COMMON/STEP/DELTAT,DELTAX,DELTAY
5040        COMMON/ARRAY/HEFFM(NX1,NY1),UVM(NX,NY)
5050        COMMON/OUTFLO/OUT(NX1,NY1)
5060        COMMON/GROWTH/FGROW(21)
5070        COMMON/STRESS/STRESS(NX1,NY1,3)
5072        COMMON/DIFF/HDF(NX1,NY1)
5073        EQUIVALENCE(HDIFF,GR)
5074        EQUIVALENCE(FHEFF,FHE1)
5075        EQUIVALENCE(FHEFF,FHE1)
5076        EQUIVALENCE(AREA,AREA1)
5080        EQUIVALENCE (HEFF,HEFF1)
5090        EQUIVALENCE (UERR,UERRV),(VERR,VERRV)
5100 * NOW DECIDE ON BASIC PARAMETERS
5110        DELTT=86400.0
5120        DELTAT=86400.0
5130        DELTAX=1.25D+05
5140        DELTAY=1 25D+05
5150        ERROR= 000001D+00
5160        ERROR=5.0D+00*ERROR
5162        FHSUM1=0.0
5163        GRSUM1=0.0
5164        ARSUM1=0.0
5170        DIFF1=.002D+00*DELTAX
5180        DIFF1=2.0*DIFF1
5190        A22=0.15D+00
5200        HO=0.5
5210 * DOUBLE HO BECAUSE OF MOD IN GROWTH
5220        HO=2.0*HO
5230        LAD=2
5240        KSTOP=1
5250 * NOW INITIALIZE COUNTER
5260        ICOUNT=0
5270        TOUT=0.0
5280        HESUM=0.0
5290        UVSUM=0.0
5300        HHESUM=0.0
5310 * NOW DEFINE BOUNDARIES
5320        CALL BNDRY
5330 *      WRITE(20) ((OUT(I,J),I=1,NX1),J=1,NY1)
5340        DO 122 J=1,NY1
5350        DO 122 I=1,NX1
5360        OUT1(I,J)=HEFFM(I,J)-OUT(I,J)
5370        HESUM=HESUM+OUT(I,J)
5380        HHESUM=HHESUM+HEFFM(I,J)
5390   122  CONTINUE
5400        DO 125 J=1,NY
5410        DO 125 I=1,NX
5420        UVSUM=UVSUM+UVM(I,J)
5430   125  CONTINUE
5440        PRINT 3,((HEFFM(I,J),I=1,NX),J=1,NY)
5450        PRINT 4
5460        PRINT 3,((UVM(I,J),I=1,NX),J=1,NY)
5470        PRINT 4
5480        PRINT 3,((OUT(I,J),I=1,NX),J=1,NY)
5490        PRINT 4
5500        PRINT 6,HESUM,UVSUM
5510        PRINT 6,HHESUM,UVSUM
5520   6    FORMAT(1X,'HESUM IS',G15.6,'UVSUM IS',G15.7)
5530 * NOW READ IN GEOSTROPHIC WIND AND CURRENTS
5540 * IF WK LT 0 THEN NEW CURRENTS READ IN. IF WK GT 100 THEN CUR ZEROED
5550 *    31 IS WIND FILE
5560 *    30 IS CURRENT FILE
5570        WK=-1.0D+00
```

```
      CALL CEO(GAIRX,GAIRY,GWATX,GWATY,WK)
      PRINT 3,((GAIRX(I,J),I=1,NX),J=1,NY)
      PRINT 4
      PRINT 3,((GAIRY(I,J),I=1,NX),J=1,NY)
      PRINT 4
      PRINT 3,((GWATX(I,J),I=1,NX),J=1,NY)
      PRINT 4
*  NOW SET UP GROWTH RATES
      CALL FGROWP
* FILE 32 CONTAINS GROWTH RATES
*     READ(32) FGRW
* NOW INITIALIZE SYSTEM
* FIRST GUESS AT INITIAL HEFF AND AREA
      DO 101 J=1,NY1
      DO 101 I=1,NX1
      HDIFF(I,J)=0.0
      HEFF(I,J,3)=0.0
      HEFF(I,J,2)=0.0
      AREA(I,J,2)=1.0
      AREA(I,J,3)=1.0
      HEFF(I,J,1)=(3.0D+00)/0.91D+00
      HEFF(I,J,1)=HEFF(I,J,1)*OUT(I,J)
      AREA(I,J,1)=1.0
  101 CONTINUE
      CALL XSUM(HEFF1,THEFF)
* NOW ZERO OUT VELOCITY
      DO 102 J=1,NY
      DO 102 I=1,NX
      UICE(I,J,3)=0.0
      UICE(I,J,2)=0.0
      UICE(I,J,1)=0.0
      VICE(I,J,3)=0.0
      VICE(I,J,2)=0.0
      VICE(I,J,1)=0.0
      UICEC(I,J)=0.0
      VICEC(I,J)=0.0
  102 CONTINUE
* NOW GET FIRST VALUE OF U AND V
      THETA=1.0
      CALL FORM(UICE,VICE,ETA,ZETA,DRAGS,DRAGA,GAIRX,GAIRY,GWATX,GWATY
     &,   FORCEX,FORCEY,HEFF,AMASS,AREA)
      PRINT 3,((FORCEX(I,J),I=1,NX),J=1,NY)
      PRINT 4
      PRINT 3,((FORCEY(I,J),I=1,NX),J=1,NY)
* NOW SET AMASS=0 AND DEFINE INITIAL VISCOSITY
      DO 103 J=1,NY
      DO 103 I=1,NX
      AMASS(I,J)=0.0
      ZETA(I,J)=HEFF(I,J,1)*(1.0D+11)
      ETA(I,J)=ZETA(I,J)/4.0
  103 CONTINUE
      CALL RELAX(UICE,VICE,ETA,ZETA,DRAGS,DRAGA,AMASS,FORCEX,FORCEY
     &,ERROR,THETA,UICEC,VICEC)
      PRINT 4
      PRINT 4
      PRINT 3,((UICE(I,J,1),I=1,NX),J=1,NY)
      PRINT 4
      PRINT 3,((VICE(I,J,1),I=1,NX),J=1,NY)
      PRINT 4
      PRINT 4
* NOW START STANDARD PREDICTOR CORRECTOR PROCEDURE
  100 CONTINUE
* NOW PUT GROWTH RATES IN
      KGRO=MOD(ICOUNT,365)+1
*     DO 141 KGG=1,21
*     FGROW(KGG)=FGRW(KGRO,KGG)/(8.64D+06)
```

```
*141  CONTINUE
* NOW SET UP REASONABLE OUTFLOW THICKNESSES
      CALL MEAN(HEFF,OUT2)
      DO 123 J=2,NY
      DO 123 I=2,NX
      HEFF(I,J,1)=HEFF(I,J,1)+OUT1(I,J)*OUT2(I,J)
 123  CONTINUE
      CALL MEAN(AREA,OUT2)
      DO 124 J=2,NY
      DO 124 I=2,NX
      AREA(I,J,1)=AREA(I,J,1)+OUT1(I,J)*OUT2(I,J)
 124  CONTINUE
      CALL XSUM(HEFF1,THEFF1)
      THEFF1=THEFF1-THEFF
* FIRST DO PREDICTOR
      DO 121 J=1,NY
      DO 121 I=1,NX
      UICE(I,J,3)=UICE(I,J,1)
      VICE(I,J,3)=VICE(I,J,1)
      UICEC(I,J)=UICE(I,J,1)
      VICEC(I,J)=VICE(I,J,1)
 121  CONTINUE
*     DO 121 J=1,NY
*     DO 121 I=1,NX
*     UICEC(I,J)=UICE(I,J,1)
*     VICEC(I,J)=VICE(I,J,1)
*121  CONTINUE
      THETA=1.0
*     DELTAT=DELTT/2.0
      DELTAT=DELTT/2.0
      CALL FORM(UICE,VICE,ETA,ZETA,DRAGS,DRAGA,GAIRX,GAIRY,GWATX,GWATY
     &,   FORCEX,FORCEY,HEFF,AMASS,AREA)
      CALL RELAX(UICE,VICE,ETA,ZETA,DRAGS,DRAGA,AMASS,FORCEX,FORCEY
     &,ERROR,THETA,UICEC,VICEC)
* NOW DO REGULAR TIME STEP
* NOW DOING BACKWARDS TIME STEP
      THETA=1.0
      DELTAT=DELTT
      CALL FORM(UICE,VICE,ETA,ZETA,DRAGS,DRAGA,GAIRX,GAIRY,GWATX,GWATY
     &,   FORCEX,FORCEY,HEFF,AMASS,AREA)
* NOW SET U(1)=U(2) AND SAME FOR V
      DO 111 J=1,NY
      DO 111 I=1,NX
      UICE(I,J,3)=UICE(I,J,1)
      VICE(I,J,3)=VICE(I,J,1)
      UICEC(I,J)=UICE(I,J,1)
      VICEC(I,J)=VICE(I,J,1)
*     UICEC(I,J)=UICE(I,J,1)
*     VICEC(I,J)=VICE(I,J,1)
      UICE(I,J,1)=UICE(I,J,2)
      VICE(I,J,1)=VICE(I,J,2)
 111  CONTINUE
      CALL RELAX(UICE,VICE,ETA,ZETA,DRAGS,DRAGA,AMASS,FORCEX,FORCEY
     &,ERROR,THETA,UICEC,VICEC)
      ICOUNT=ICOUNT+1
* NOW DO ADVECTION
      CALL ADVECT(UICE,VICE,HEFF,DIFF1,LAD)
      CALL ADVECT(UICE,VICE,AREA,DIFF1,LAD)
* NOW DO GROWTH
      CALL GROWTH(HEFF,AREA,HCORR,HO,A22)
*     MUST CALL GROWTH ONLY AFTER CALLING ADVECTION
* NOW CORRECT OUTFLOW PTS AND GET OUTFLOW ICE
      CALL XSUM(HEFF1,THEFF)
      DO 105 J=1,NY1
      DO 105 I=1,NX1
      HEFF(I,J,1)=HEFF(I,J,1)*OUT(I,J)
```

```
6880        AREA(I,J,1)=AREA(I,J,1)*OUT(I,J)
6885        FHEFF(I,J)=FHEFF(I,J)*OUT(I,J)
6886        HDIFF(I,J)=(1.0-AREA(I,J,2))*FO(I,J)*OUT(I,J)*DELTT
6890  105   CONTINUE
6900        CALL XSUM(HEFF1,THEFF2)
6902        CALL XSUM(AREA1,ARSUM)
6903        CALL XSUM(FHE1,FHSUM)
6904        CALL XSUM(GR,GRSUM)
6905        GRSUM1=GRSUM1+GRSUM
6906        FHSUM1=FHSUM1+FHSUM
6907        ARSUM1=ARSUM1+ARSUM
6910        TOUT1=THEFF-THEFF2-THEFF1
6920        THEFF=THEFF2
6930        TOUT=TOUT+TOUT1
6935        ICOU=ICOUNT
6940 * NOW PRINT OUT EVERY KTH PT
6945 *      IF(ICOUNT.LT.366) GO TO 97
6946 *      ICOU=ICOUNT-365
6950        KTH=MOD(ICOUNT,8)
6955        KTH=MOD(ICOUNT,20)
6960        IF(KTH.EQ.1) GO TO 95
6970        GO TO 96
6980  95    CONTINUE
6990  4     FORMAT(//)
7000  1     FORMAT(1X,'TIME STEP AND TOTAL THICKNESS ARE',2G20.12)
7010        PRINT 1,ICOUNT,THEFF1
7020        PRINT 1,ICOUNT,THEFF
7030        PRINT 4
7040        PRINT 2,TOUT1,TOUT
7050        PRINT 4
7060  2     FORMAT(1X,'OUTFLOW AND NET OUTFLOW ARE',2G14.7)
7070        PRINT 3,((UICE(I,J,1),I=1,NX),J=1,NY)
7080        PRINT 4
7090        PRINT 3,((VICE(I,J,1),I=1,NX),J=1,NY)
7100        PRINT 4
7110        PRINT 3,((HEFF(I,J,1),I=1,NX),J=1,NY)
7120        PRINT 4
7130        PRINT 3,((AREA(I,J,1),I=1,NX),J=1,NY)
7140        PRINT 4
7150        PRINT 3,((ZETA(I,J),I=1,NX),J=1,NY)
7160        PRINT 4
7170        PRINT 3,((FHEFF(I,J),I=1,NX),J=1,NY)
7180        PRINT 4
7190        PRINT 3,((STRESS(I,J,1),I=1,NX),J=1,NY)
7200        PRINT 4
7210        PRINT 3,(FGROW(I),I=1,21)
7220        PRINT 4
7230  3     FORMAT(1X,5G20.14)
7240 *  NOW PRINT OUT SQUARE VELOCITY AND VELOCITY DIFERENCE AND MAX CHANGE
7250        SQ1=0.0
7260        SQ=0.0
7270        DO 130 J=1,NY
7280        DO 130 I=1,NX
7290        SQ=SQ+UICE(I,J,1)**2+VICE(I,J,1)**2
7300        UERR(I,J)=UICE(I,J,1)-UICE(I,J,2)
7310        VERR(I,J)=VICE(I,J,1)-VICE(I,J,2)
7320        SQ1=SQ1+UERR(I,J)**2+VERR(I,J)**2
7330  130   CONTINUE
7340 *      ISU=MAXMAG(UERRV)
7350 *      ISV=MAXMAG(VERRV)
7355 *      SMU=DABS(UERRV(ISU+1))
7356 *      SMV=DABS(VERRV(ISV+1))
7360        CALL XMAXM(UERRV,SMU)
7370        CALL XMAXM(VERRV,SMV)
7380        SM=DMAX1(SMU,SMV)
7390        PRINT 5,SQ,SQ1,SM
```

```
 5    FORMAT(1X,'SQUARE VELOCITY,SQ. VELOCITY DIFFERENCE,MAX CHANGE
     &/1X,3G20.12)
      PRINT 4
 96   CONTINUE
*     WRITE(20) ((UICE(I,J,1),I=1,NX),J=1,NY)
*     WRITE(20) ((VICE(I,J,1),I=1,NX),J=1,NY)
*     WRITE (20) ((HEFF(I,J,1),I=1,NX1),J=1,NY1)
*     WRITE (20) ((AREA(I,J,1),I=1,NX1),J=1,NY1)
*     WRITE (20) ((FHEFF(I,J,1),I=1,NX1),J=1,NY1)
*     WRITE (20) ((ZETA(I,J),I=1,NX1),J=1,NY1)
*     WRITE (20) ((STRESS(I,J,1),I=1,NX1),J=1,NY1)
*     WRITE(23) TOUT1,FHSUM,GRSUM,ARSUM,THEFF
 97   CONTINUE
      PRINT 1,ICOU,THEFF
      PRINT 2,TOUT1,TOUT
      PRINT 8,FHSUM,FHSUM1
      PRINT 9,GRSUM,GRSUM1,ARSUM,ARSUM1
 9    FORMAT(1X,'OPEN WATER GROWTH AND NET GROWTH ETC',4G15.7)
 8    FORMAT(1X,'GROWTH AND NET GROWTH ARE',2G14.7)
* NOW CHECK COUNTER AND DECIDE IF NEW WINDS NEEDED
      KTH1=MOD(ICOUNT,365)
      IF(KTH1.EQ.0) GO TO 211
      GO TO 212
 211  CONTINUE
*     REWIND 31
      PRINT 7
 7    FORMAT (1X,'WIND FILE REWOUND')
 212  CONTINUE
*     READ(31) GAIRX,GAIRY
* NOW DECIDE IF DONE
      IF(ICOUNT.GT.50) GO TO 200
      IF(ICOUNT.GT.20) GO TO 200
      GO TO 100
 200  CONTINUE
      KSTOP=2
      IF (KSTOP.EQ.1) GO TO 202
      GO TO 201
 202  CONTINUE
      DO 204 J=1,NY
      DO 204 I=1,NX
      GAIRX(I,J)=0.0
      GAIRY(I,J)=0.0
 204  CONTINUE
      ICOUNT=0
      KSTOP=2
      GO TO 100
 201  CONTINUE
*     WRITE(21) UICE,VICE,HEFF,AREA
      STOP
      END
      SUBROUTINE MEAN(HEFF,HMEAN)
* SUBROUTINE FINDS MEAN HEFF AT OUTFLOW PTS OF VALUES AROUND
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1,N4=NX*NY
      DIMENSION HEFF(NX1,NY1,3),HMEAN(NX1,NY1)
      COMMON/OUTFLO/OUT(NX1,NY1)
      DO 101 J=2,NY
      DO 101 I=2,NX
      HMEAN(I,J)=(HEFF(I+1,J,1)*OUT(I+1,J)+HEFF(I+1,J+1,1)*OUT(I+1,J+1)
     & +HEFF(I+1,J-1,1)*OUT(I+1,J-1)+HEFF(I,J+1,1)*OUT(I,J+1)
     & +HEFF(I,J-1,1)*OUT(I,J-1)+HEFF(I-1,J,1)*OUT(I-1,J)
     &  +HEFF(I-1,J+1,1)*OUT(I-1,J+1)+HEFF(I-1,J-1,1)*OUT(I-1,J-1)
     &)/(OUT(I+1,J)+OUT(I+1,J+1)+OUT(I+1,J-1)+OUT(I,J+1)+OUT(I,J-1)
     &  +OUT(I-1,J)+OUT(I-1,J+1)+OUT(I-1,J-1)+.00001D+00)
 101  CONTINUE
```

```
      RETURN
      END
      SUBROUTINE PLAST(UICE,VICE,PRESS,ETA,ZETA,ECCEN,ZMAX,ZMIN)
* SUBROUTINE CALCULATES STRAIN RATES AND VISCOUS PARAMETERS
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1,N4=NX*NY
      DIMENSION UICE(NX,NY,3),VICE(NX,NY,3),PRESS(NX1,NY1)
     &,ETA(NX1,NY1),ZETA(NX1,NY1),E11(NX1,NY1),E22(NX1,NY1)
     &,E12(NX1,NY1),ZMAX(NX1,NY1),ZMIN(NX1,NY1)
      COMMON/STEP/DELTAT,DELTAX,DELTAY
      COMMON/STRESS/STRESS(NX1,NY1,3)
      COMMON/ARRAY/HEFFM(NX1,NY1),UVM(NX,NY)
      ECM2=1.0/(ECCEN**2)
      GMIN=1.0D-20
* NOW EVALUATE STRAIN RATES
      DO 101 J=2,NY
      DO 101 I=2,NX
      E11(I,J)=(0.5/DELTAX)*(UICE(I,J,1)+UICE(I,J-1,1)
     &  -UICE(I-1,J,1)-UICE(I-1,J-1,1))
      E22(I,J)=(0.5/DELTAY)*(VICE(I,J,1)+VICE(I-1,J,1)
     &  -VICE(I,J-1,1)-VICE(I-1,J-1,1))
      E12(I,J)=(0.25/DELTAY)*(UICE(I,J,1)+UICE(I-1,J,1)
     &    -UICE(I,J-1,1)-UICE(I-1,J-1,1))
     &  +(0.25/DELTAX)*(VICE(I,J,1)+VICE(I,J-1,1)
     &  -VICE(I-1,J,1)-VICE(I-1,J-1,1))
*  NOW EVALUATE VISCOSITIES
      DELT=(E11(I,J)**2+E22(I,J)**2)*(1.0+ECM2)+4.0*ECM2*E12(I,J)**2
     &  +2.0*E11(I,J)*E22(I,J)*(1.0-ECM2)
      DELT1=DSQRT(DELT)
      DELT1=DMAX1(GMIN,DELT1)
      ZETA(I,J)=0.5*PRESS(I,J)/DELT1
 101  CONTINUE
* NOW PUT MIN AND MAX VISCOSITIES IN
      DO 102 J=1,NY1
      DO 102 I=1,NX1
      ZETA(I,J)=DMIN1(ZMAX(I,J),ZETA(I,J))
      ZETA(I,J)=DMAX1(ZMIN(I,J),ZETA(I,J))
      ETA(I,J)=ECM2*ZETA(I,J)
      E11(I,J)=E11(I,J)*HEFFM(I,J)
      E22(I,J)=E22(I,J)*HEFFM(I,J)
      E12(I,J)=E12(I,J)*HEFFM(I,J)
      SS11=(ZETA(I,J)-ETA(I,J))*(E11(I,J)+E22(I,J))-PRESS(I,J)*0.5
      STRESS(I,J,1)=(2.0*ETA(I,J)*E11(I,J)+SS11)
      STRESS(I,J,2)=2.0*ETA(I,J)*E22(I,J)+SS11
      STRESS(I,J,3)=2.0*ETA(I,J)*E12(I,J)
 102  CONTINUE
      RETURN
      END
      SUBROUTINE RELAX(OUICE,OVICE,OETA,OZETA,ODRAGS,ODRAGA
     &,OAMASS,ORCEX,ORCEY,ORROR,OTHETA,OUICEC,OVICEC)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1,N4=NX*NY
      DIMENSION UICE(NX,NY,3),VICE(NX,NY,3),ETA(NX1,NY1),ZETA(NX1,NY1)
     &,DRAGS(NX,NY),DRAGA(NX,NY),FORCEX(NX,NY),FORCEY(NX,NY),FXETA(4)
     &,FXZETA(4),FYETA(4),FYZETA(4),UERR(NX,NY),UERRV(N4)
     &,VERR(NX,NY),VERRV(N4),COEF(NX,NY),AMASS(NX,NY)
     &,FXM(NX,NY),FYM(NX,NY)
     &,UICEC(NX,NY),VICEC(NX,NY)
     &,FXE(4,NX,NY),FYE(4,NX,NY),FXZ(4,NX,NY),FYZ(4,NX,NY)
     &,COEFI(NX,NY)
     &,OUICE(NX,NY,3),OVICE(NX,NY,3),OETA(NX1,NY1),OZETA(NX1,NY1)
     &,ODRAGS(NX,NY),ODRAGA(NX,NY),OAMASS(NX,NY),ORCEX(NX,NY)
     &,ORCEY(NX,NY),OUICEC(NX,NY),OVICEC(NX,NY)
     &,HEFFM(NX1,NY1),UVM(NX,NY)
```

```
      COMMON/DINV/DELIN2
      COMMON/STEP/OTAT, OTAX, OTAY
      COMMON/ARRAY/OHEFFM(NX1, NY1), OUVM(NX, NY)
      EQUIVALENCE (UERR, UERRV), (VERR, VERRV)
      ICOUNT=0
      WFA=1. 5
* NOW REPLACE DOUBLE PRECISION INPUT BY SINGLE PRECISION VARIABLES
      DO 601 J=1, NY
      DO 601 I=1, NX
      UVM(I, J)=OUVM(I, J)
      DRAGS(I, J)=ODRAGS(I, J)
      DRAGA(I, J)=ODRAGA(I, J)
      AMASS(I, J)=OAMASS(I, J)
      FORCEX(I, J)=ORCEX(I, J)
      FORCEY(I, J)=ORCEY(I, J)
      UICEC(I, J)=OUICEC(I, J)
      VICEC(I, J)=OVICEC(I, J)
  601 CONTINUE
      DO 603 K=1, 3
      DO 603 J=1, NY
      DO 603 I=1, NX
      UICE(I, J, K)=OUICE(I, J, K)
      VICE(I, J, K)=OVICE(I, J, K)
  603 CONTINUE
      DO 602 J=1, NY1
      DO 602 I=1, NX1
*     HEFFM(I, J)=OHEFFM(I, J)
      ETA(I, J)=OETA(I, J)
      ZETA(I, J)=OZETA(I, J)
  602 CONTINUE
      ERROR=ORROR
      THETA=OTHETA
      DELTAX=OTAX
*     DELTAY=OTAY
      DELTAT=OTAT
      DELIN =1. 0/DELTAX
      DELIN2=0. 5/(DELTAX**2)
      K=1
* MUST UPDATE HEFF BEFORE CALLING RELAC
* FIRST SET U(2)=U(1)
      DO 99 J=1, NY
      DO 99 I=1, NX
* NOW MAKE SURE BDRY PTS ARE EQUAL TO ZERO
      UICE(I, J, 2)=UICE(I, J, 1)
      VICE(I, J, 2)=VICE(I, J, 1)
      OUICE(I, J, 2)=OUICE(I, J, 1)
      OVICE(I, J, 2)=OVICE(I, J, 1)
      OUICE(I, J, 1)=OUICE(I, J, 3)*OUVM(I, J)
      OVICE(I, J, 1)=OVICE(I, J, 3)*OUVM(I, J)
      UICE(I, J, 1)=UICE(I, J, 3)*UVM(I, J)
      VICE(I, J, 1)=VICE(I, J, 3)*UVM(I, J)
   99 CONTINUE
* NOW SET UP COEFFICIENTS OF DIAGONAL COMPONENTS
  501 CONTINUE
      DO 102 J=1, NY
      DO 102 I=1, NX
      COEF(I, J)=AMASS(I, J)/DELTAT+2. 0*THETA*(0. 5*DRAGS(I, J)
     &+2. 0*((ETA(I, J)+ETA(I+1, J)+ETA(I, J+1)+ETA(I+1, J+1))
     &+. 5*(ZETA(I, J)+ZETA(I+1, J)+ZETA(I, J+1)+ZETA(I+1, J+1))
     &   )/(4. 0*(DELTAX**2)))
      COEFI(I, J)=1. 0/COEF(I, J)
  102 CONTINUE
* NOW CALCULATE ALL FUNCTIONS OF PREVIOUS U AND V VALUES
      TTHETA=2. 0*(1. 0-THETA)
      DO 111 J=2, NYM1
      DO 111 I=2, NXM1
```

```
502  CONTINUE
     CALL FELLIP(UICE,VICE,ETA,FXETA,I,J,2)
     CALL FELLIP(UICE,VICE,ZETA,FXZETA,I,J,2)
     CALL FELLIP(VICE,UICE,ETA,FYETA,I,J,2)
     CALL FELLIP(VICE,UICE,ZETA,FYZETA,I,J,2)
     FX0=0.5*(FXETA(1)+FXZETA(1)+FXETA(2)+FXETA(3)+FXZETA(4)-FXETA(4))
     FX0=TTHETA*FX0
     FX1=(AMASS(I,J)/DELTAT-TTHETA*0.5*DRAGS(I,J))*UICE(I,J,2)
     FX2=TTHETA*0.5*DRAGA(I,J)*VICE(I,J,2)
     FY0=0.5*(FYETA(1)+FYETA(2)+FYZETA(2)+FYZETA(3)-FYETA(3)+FYETA(4))
     FY0=FY0*TTHETA
     FY1=(AMASS(I,J)/DELTAT-TTHETA*0.5*DRAGS(I,J))*VICE(I,J,2)
     FY2=-TTHETA*0.5*DRAGA(I,J)*UICE(I,J,2)
     FXC=AMASS(I,J)*0.5*TTHETA*
    &  (UICEC(I,J)*(UICE(I+1,J,2)-UICE(I-1,J,2))
    &  +VICEC(I,J)*(UICE(I,J+1,2)-UICE(I,J-1,2)))/(2.0*DELTAX)
     FXM(I,J)=FX0+FX1+FX2+FORCEX(I,J)+FXC
      FYC=AMASS(I,J)*0.5*TTHETA*
    &    (UICEC(I,J)*(VICE(I+1,J,2)-VICE(I-1,J,2))
    &   +VICEC(I,J)*(VICE(I,J+1,2)-VICE(I,J-1,2)))/(2.0*DELTAX)
      FYM(I,J)=FY0+FY1+FY2+FORCEY(I,J)+FYC
 111  CONTINUE
* NOW SET U(3)=U(1)
 100  CONTINUE
      DO 101 J=1,NY
      DO 101 I=1,NX
      UICE(I,J,3)=UICE(I,J,1)
      VICE(I,J,3)=VICE(I,J,1)
 101  CONTINUE
* NOW BEGIN SWEEP
      CALL FELLD1(UICE,VICE,ETA,FXE,1)
      CALL FELLD1(UICE,VICE,ZETA,FXZ,1)
      CALL FELLD1(VICE,UICE,ETA,FYE,1)
      CALL FELLD1(VICE,UICE,ZETA,FYZ,1)
      DO 103 J=2,NYM1
      DO 103 I=2,NXM1
 504  CONTINUE
      K=1
      FXETA(1) = FXE(1,I,J)+DELIN2*
    &(UICE(I-1,J,K)*( ETA(I,J+1)+ ETA(I,J)))
      FXETA(2)=FXE(2,I,J)+DELIN2*
    &(UICE(I,J-1,K)*(ETA(I,J)+ETA(I+1,J)))
      FXETA(3)=FXE(3,I,J)+0.5*DELIN2*
    &(VICE(I-1,J-1,K)*ETA(I,J)+VICE(I,J-1,K)*
    &(-ETA(I,J)+ETA(I+1,J))-VICE(I+1,J-1,K)*ETA(I+1,J)
    &+VICE(I-1,J,K)*(-ETA(I,J+1)+ETA(I,J))
    &-VICE(I-1,J+1,K)*ETA(I,J+1))
      FXETA(4)=FXE(4,I,J)+DELIN2*0.5*
    &(VICE(I-1,J-1,K)*ETA(I,J)+VICE(I,J-1,K)*
    &(-ETA(I+1,J)+ETA(I,J))-VICE(I+1,J-1,K)*ETA(I+1,J)
    &+VICE(I-1,J,K)*(ETA(I,J+1)-ETA(I,J))
    &-VICE(I-1,J+1,K)*ETA(I,J+1))
      FYETA(1)=FYE(1,I,J)+DELIN2*
    &(VICE(I-1,J,K)*( ETA(I,J+1)+ ETA(I,J)))
      FYETA(2)=FYE(2,I,J)+DELIN2*
    &(VICE(I,J-1,K)*(ETA(I,J)+ETA(I+1,J)))
      FYETA(3)=FYE(3,I,J)+0.5*DELIN2*
    &(UICE(I-1,J-1,K)*ETA(I,J)+UICE(I,J-1,K)*
    &(-ETA(I,J)+ETA(I+1,J))-UICE(I+1,J-1,K)*ETA(I+1,J)
    &+UICE(I-1,J,K)*(-ETA(I,J+1)+ETA(I,J))
    &-UICE(I-1,J+1,K)*ETA(I,J+1))
      FYETA(4)=FYE(4,I,J)+DELIN2*0.5*
    &(UICE(I-1,J-1,K)*ETA(I,J)+UICE(I,J-1,K)*
    &(-ETA(I+1,J)+ETA(I,J))-UICE(I+1,J-1,K)*ETA(I+1,J)
    &+UICE(I-1,J,K)*(ETA(I,J+1)-ETA(I,J))
    &-UICE(I-1,J+1,K)*ETA(I,J+1))
```

```
      FXZETA(1)=FXZ(1, I, J)+DELIN2*
     &(UICE(I-1, J, K)*(ZETA(I, J+1)+ZETA(I, J)))
      FXZETA(4)=FXZ(4, I, J)+DELIN2*0. 5*
     &(VICE(I-1, J-1, K)*ZETA(I, J)+VICE(I, J-1, K)*
     &(-ZETA(I+1, J)+ZETA(I, J))-VICE(I+1, J-1, K)*ZETA(I+1, J)
     &+VICE(I-1, J, K)*(ZETA(I, J+1)-ZETA(I, J))
     &-VICE(I-1, J+1, K)*ZETA(I, J+1))
      FYZETA(2)=FYZ(2, I, J)+DELIN2*
     &(VICE(I, J-1, K)*(ZETA(I, J)+ZETA(I+1, J)))
      FYZETA(3)=FYZ(3, I, J)+0. 5*DELIN2*
     &(UICE(I-1, J-1, K)*ZETA(I, J)+UICE(I, J-1, K)*
     &(-ZETA(I, J)+ZETA(I+1, J))-UICE(I+1, J-1, K)*ZETA(I+1, J)
     &+UICE(I-1, J, K)*(-ZETA(I, J+1)+ZETA(I, J))
     &-UICE(I-1, J+1, K)*ZETA(I, J+1))
      FX3=0. 5*(FXETA(1)+FXZETA(1)+FXETA(2)+FXETA(3)+FXZETA(4)-FXETA(4))
      FX3=FX3*2. 0*THETA
      FXCP=AMASS(I, J)*THETA*
     &  (UICEC(I, J)*(UICE(I+1, J, 1)-UICE(I-1, J, 1))
     &+VICEC(I, J)*(UICE(I, J+1, 1)-UICE(I, J-1, 1)))*0. 5*DELIN
      FX3=FX3+FXCP
      FY3=0. 5*(FYETA(1)+FYETA(2)+FYZETA(2)+FYZETA(3)-FYETA(3)+FYETA(4))
      FY3=FY3*2. 0*THETA
      FYCP=AMASS(I, J)*THETA*
     &  (UICEC(I, J)*(VICE(I+1, J, 1)-VICE(I-1, J, 1))
     &+VICEC(I, J)*(VICE(I, J+1, 1)-VICE(I, J-1, 1)))*0. 5*DELIN
      FY3=FY3+FYCP
      FL11=0. 5*DRAGA(I, J)*COEFI(I, J)
      FL11=FL11*2. 0*THETA
      F11=(FXM(I, J)+FX3)*COEFI(I, J)
      F22=(FYM(I, J)+FY3)*COEFI(I, J)
  505 CONTINUE
      FL11S=1. 0+FL11**2
      FL11SI=1. 0/FL11S
      UICOR=((F11+FL11*F22)*FL11SI)*UVM(I, J)
      VICOR=((F22-FL11*F11)*FL11SI)*UVM(I, J)
      OUICE(I, J, 1)=OUICE(I, J, 1)+WFA*(UICOR-UICE(I, J, 1))
      OVICE(I, J, 1)=OVICE(I, J, 1)+WFA*(VICOR-VICE(I, J, 1))
*     OUICE(I, J, 1)=OUICE(I, J, 1)+1. 5*(UICOR-UICE(I, J, 1))
*     OVICE(I, J, 1)=OVICE(I, J, 1)+1. 5*(VICOR-VICE(I, J, 1))
      UICE(I, J, 1)=OUICE(I, J, 1)
      VICE(I, J, 1)=OVICE(I, J, 1)
*     UICE(I, J, 1)=UICE(I, J, 1)+1. 5*(UICOR-UICE(I, J, 1))
*     VICE(I, J, 1)=VICE(I, J, 1)+1. 5*(VICOR-VICE(I, J, 1))
  103 CONTINUE
      ICOUNT=ICOUNT+1
      IF(ICOUNT. GT. 800) GO TO 201
      IF(ICOUNT. LT. 100) GO TO 202
      WFA=1. 0
  202 CONTINUE
* NOW CHECK MAX ERROR
*     S11=0. 0
*     S22=0. 0
* FORM ERROR MATRIX
      DO 104 J=1, NY
      DO 104 I=1, NX

      UERR(I, J)=UICE(I, J, 1)-UICE(I, J, 3)
      VERR(I, J)=VICE(I, J, 1)-VICE(I, J, 3)
  104 CONTINUE
* NOW FIND ERROR
      CALL XMAXM(UERRV, S1)
      CALL XMAXM(VERRV, S2)
*     S1=ABS(UERRV(IU+1))
*     S2=ABS(VERRV(IV+1))
      S1=DMAX1(S1, S2)
      IF(S1. LT. ERROR) GO TO 200
```

```
       GO TO 100
  201  CONTINUE
       PRINT 11
  11   FORMAT(1X,'NO CONVERGENCE AFTER 800 ITERATIONS')
* NOW  END
  200  CONTINUE
*      PRINT 1,ICOUNT
       PRINT 12,S1
       PRINT 1,ICOUNT
  12   FORMAT(1X,'MAX ERROR AND U AND V POWER',3G12.5)
  1    FORMAT(1X,'NO OF ITERATIONS ARE',1G12.5)
       RETURN
       END
       SUBROUTINE FELLIP(UICE,VICE,ETA,F,I,J,K)
* SPACER
       IMPLICIT DOUBLE PRECISION (A-H,O-Z)
       PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
      &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1
       DIMENSION UICE(NX,NY,3),VICE(NX,NY,3),ETA(NX1,NY1),F(4)
       COMMON/STEP/OTAT,OTAX,OTAY
       DELTAX=OTAX
       S1=.5/(DELTAX**2)
       F(1)=S1*(UICE(I+1,J,K)*(ETA(I+1,J+1)+ETA(I+1,J))
      &-UICE(I,J,K)*(ETA(I+1,J+1)+ETA(I,J)+ETA(I+1,J)+ETA(I,J+1))
      &+UICE(I-1,J,K)*(ETA(I,J+1)+ETA(I,J)))
       F(2)=S1*(UICE(I,J+1,K)*(ETA(I+1,J+1)+ETA(I,J+1))
      &-UICE(I,J,K)*(ETA(I+1,J+1)+ETA(I,J)+ETA(I+1,J)+ETA(I,J+1))
      &+UICE(I,J-1,K)*(ETA(I,J)+ETA(I+1,J)))
       F(3)=S1*(VICE(I-1,J-1,K)*ETA(I,J)+VICE(I,J-1,K)*(-ETA(I,J)
      &+ETA(I+1,J))-VICE(I+1,J-1,K)*ETA(I+1,J)+VICE(I-1,J,K)*(-ETA(I,J+1)
      &+ETA(I,J))  +VICE(I,J,K)*(-ETA(I,J)-ETA(I+1,J+1)+ETA(I+1,J)
      &+ETA(I,J+1)))
       F(3)=F(3)+S1*(VICE(I+1,J,K)*(-ETA(I+1,J)+ETA(I+1,J+1))
      &-VICE(I-1,J+1,K)*ETA(I,J+1)
      &+VICE(I,J+1,K)*(-ETA(I+1,J+1)+ETA(I,J+1))
      &+VICE(I+1,J+1,K)*ETA(I+1,J+1))
       F(4)=S1*(VICE(I-1,J-1,K)*ETA(I,J)+VICE(I,J-1,K)*(-ETA(I+1,J)
      &+ETA(I,J))-VICE(I+1,J-1,K)*ETA(I+1,J)+VICE(I-1,J,K)*(ETA(I,J+1)
      &-ETA(I,J))+VICE(I,J,K)*(ETA(I,J+1)+ETA(I+1,J)-ETA(I,J)-ETA(I+1,
      &J+1)))
       F(4)=F(4)+S1*(VICE(I+1,J,K)*(ETA(I+1,J)-ETA(I+1,J+1))
      &-VICE(I-1,J+1,K)*ETA(I,J+1)
      &+VICE(I,J+1,K)*(ETA(I+1,J+1)-ETA(I,J+1))
      &+VICE(I+1,J+1,K)*ETA(I+1,J+1))
       F(3)=F(3)*.5
       F(4)=F(4)*.5
       RETURN
       END
       SUBROUTINE FELLD1(UICE,VICE,ETA,F,K)
* SPACER
       IMPLICIT DOUBLE PRECISION (A-H,O-Z)
       PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
      &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1
       DIMENSION UICE(NX,NY,3),VICE(NX,NY,3),ETA(NX1,NY1),F(4,NX,NY)
*      COMMON/STEP/DELTAT,DELTAX,DELTAY
       COMMON/DINV/S1
       DO 101 J=2,NYM1
       DO 101 I=2,NXM1
       F(1,I,J)=S1*(UICE(I+1,J,K)*(ETA(I+1,J+1)+ETA(I+1,J)))
       F(2,I,J)=S1*(UICE(I,J+1,K)*(ETA(I+1,J+1)+ETA(I,J+1)))
       F(3,I,J)=S1*(VICE(I,J,K)*(-ETA(I,J)-ETA(I+1,J+1)+ETA(I+1,J)
      &+ETA(I,J+1))
      &+VICE(I+1,J,K)*(-ETA(I+1,J)+ETA(I+1,J+1))
      &+VICE(I,J+1,K)*(-ETA(I+1,J+1)+ETA(I,J+1))
      &+VICE(I+1,J+1,K)*ETA(I+1,J+1))
       F(4,I,J)=S1*(VICE(I,J,K)*(ETA(I,J+1)+ETA(I+1,J)-ETA(I,J)-ETA(I+1,
     &J+1))+VICE(I+1,J,K)*(ETA(I+1,J)-ETA(I+1,J+1))
     &+VICE(I,J+1,K)*(ETA(I+1,J+1)-ETA(I,J+1))
     &+VICE(I+1,J+1,K)*ETA(I+1,J+1))
      F(3,I,J)=0.5*F(3,I,J)
      F(4,I,J)=0.5*F(4,I,J)
  101 CONTINUE
      RETURN
      END
      SUBROUTINE FELLD(UICE,VICE,ETA,F,I,J,K,FETA)
* SPACER
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1
      DIMENSION UICE(NX,NY,3),VICE(NX,NY,3),ETA(NX1,NY1),F(4)
     &,FETA(4,NX,NY)
      COMMON/DINV/S1
      F(1)=FETA(1,I,J)+S1*(
     &+UICE(I-1,J,K)*(ETA(I,J+1)+ETA(I,J)))
      F(2)=FETA(2,I,J)+S1*(
     &+UICE(I,J-1,K)*(ETA(I,J)+ETA(I+1,J)))
      F(3)=FETA(3,I,J)+0.5*S1*(VICE(I-1,J-1,K)*ETA(I,J)+VICE(I,J-1,K)*
     &(-ETA(I,J)
     &+ETA(I+1,J))-VICE(I+1,J-1,K)*ETA(I+1,J)+VICE(I-1,J,K)*(-ETA(I,J+1)
     &+ETA(I,J))
     &-VICE(I-1,J+1,K)*ETA(I,J+1))
      F(4)=FETA(4,I,J)+S1*0.5*
     &(VICE(I-1,J-1,K)*ETA(I,J)+VICE(I,J-1,K)*(-ETA(I+1,J)
     &+ETA(I,J))-VICE(I+1,J-1,K)*ETA(I+1,J)+VICE(I-1,J,K)*(ETA(I,J+1)
     &-ETA(I,J))
     &-VICE(I-1,J+1,K)*ETA(I,J+1))
      RETURN
      END
      SUBROUTINE XMAXM(UERRV,X)
*     SPACER
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1,N4=NX*NY
      DIMENSION UERRV(N4)
      X=0.0
      DO 100 I=1,N4
      X=DMAX1(DABS(UERRV(I)),X)
  100 CONTINUE
      RETURN
      END
      SUBROUTINE XSUM(HEFF1,S1)
* PROGRAM SUMS UP VECTOR
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER NX=5,NY=6,N3=(NX+1)*(NY+1),NX1=NX+1
     &,NY1=NY+1,NXM1=NX-1,NYM1=NY-1
      DIMENSION HEFF1(N3)
      S1=0.0
      DO 100 I=1,N3
      S1=S1+HEFF1(I)
  100 CONTINUE
      RETURN
      END
```

APPENDIX B: TWENTY-ONE-DAY TEST RUN OF MODEL

This 21-day test run is obtained by compiling and executing the code as listed in Appendix A. The only input data needed are contained in a file BNDATA which is read by subroutine BNDRY (lines 1020-1050). This BNDATA file consists of the following formatted data which define the boundary conditions shown in Figure 2:

```
000000111001110011100111000000
000000011110011110011110011110011110000000
000000011100011110011110011110011110000000
```

```
INTEGER VARIABLE (K2) SET, NEVER USED, LINES 250 AND 300

DOUBLE PRECISION VARIABLE (DWAT) SET, NEVER USED, LINE 2660
DOUBLE PRECISION VARIABLE (DAIR) SET, NEVER USED, LINE 2670.
DOUBLE PRECISION VARIABLE (GRAV) SET, NEVER USED, LINE 2750
DOUBLE PRECISION VARIABLE (EPSIL) SET, NEVER USED, LINE 3210.
DOUBLE PRECISION VARIABLE (ETAC) SET, NEVER USED, LINE 3240.

DOUBLE PRECISION VARIABLE (ONE) SET, NEVER USED, LINE 3890
DOUBLE PRECISION VARIABLE (ZERO) SET, NEVER USED, LINE 3900.

INTEGER VARIABLE (KGRO) SET, NEVER USED, LINE 6200

  00000000000000      00000000000000     .00000000000000      00000000000000      00000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000


  00000000000000      00000000000000      00000000000000      00000000000000      00000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000      00000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000      00000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000      00000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000      00000000000000
  00000000000000      00000000000000     .00000000000000      00000000000000      00000000000000


  00000000000000      00000000000000      00000000000000     .00000000000000      00000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000      00000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000
  00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000


HESUM IS     19.0000    UVSUM IS    12.00000
HESUM IS     20.0000    UVSUM IS    12.00000
 3.2000000000000     3.3000000000000     3.4000000000000     3.5000000000000     3.6000000000000
 3.3000000000000     3.4000000000000     3.5000000000000     3.6000000000000     3.7000000000000
 3.4000000000000     3.5000000000000     3.6000000000000     3.7000000000000     3.8000000000000
 3.5000000000000     3.6000000000000     3.7000000000000     3.8000000000000     3.9000000000000
 3.6000000000000     3.7000000000000     3.8000000000000     3.9000000000000     4.0000000000000
 3.7000000000000     3.8000000000000     3.9000000000000     4.0000000000000     4.1000000000000


-3.6000000000000    -3.4000000000000    -3.2000000000000    -3.0000000000000    -2.8000000000000
-3.4000000000000    -3.2000000000000    -3.0000000000000    -2.8000000000000    -2.6000000000000
-3.2000000000000    -3.0000000000000    -2.8000000000000    -2.6000000000000    -2.4000000000000
-3.0000000000000    -2.8000000000000    -2.6000000000000    -2.4000000000000    -2.2000000000000
-2.8000000000000    -2.6000000000000    -2.4000000000000    -2.2000000000000    -2.0000000000000
-2.6000000000000    -2.4000000000000    -2.2000000000000    -2.0000000000000    -1.8000000000000


  .10000000000000E-01  .90000000000000E-02  .80000000000000E-02  .70000000000000E-02  .60000000000000E-02
  .11000000000000E-01  .10000000000000E-01  .90000000000000E-02  .80000000000000E-02  .70000000000000E-02
  .12000000000000E-01  .11000000000000E-01  .10000000000000E-01  .90000000000000E-02  .80000000000000E-02
  .13000000000000E-01  .12000000000000E-01  .11000000000000E-01  .10000000000000E-01  .90000000000000E-02
  .14000000000000E-01  .13000000000000E-01  .12000000000000E-01  .11000000000000E-01  .10000000000000E-01
  .15000000000000E-01  .14000000000000E-01  .13000000000000E-01  .12000000000000E-01  .11000000000000E-01
```

```
     1    .139930555556E-05
     2    .225694444444E-06
     3    .532407407407E-07
     4    .428240740741E-07
     5    .358796296296E-07
     6    .312500000000E-07
     7    .243055555556E-07
     8    .162037037037E-07
     9    .104166666667E-07
    10    .347222222222E-08
    11    .000000000000
    12   -.231481481481E-08
    13   -.347222222222E-08
    14   -.462962962963E-08
    15   -.578703703704E-08
    16   -.578703703704E-08
    17   -.694444444444E-08
    18   -.694444444444E-08
    19   -.694444444444E-08
    20   -.694444444444E-08
    21   -.694444444444E-08


-.919848552431 25E-02 .169091332384 02E-01 .157933299425 52E-01 .537194274077 23E-01 .259853532443 02E-01
```

```
-.91984855243125E-02 .16909133238402E-01 .15793329942552E-01 .53719427407723E-01 .25985353244302E-01
-.48514339437489E-01 .44866731198317E-02 .32137460399924E-02 .41095469141595E-01 .52528457534129E-01
-.49100120038751E-01 .37432349310818E-02 .25349005658436E-02 .13850900150383E-02 .78931630880927E-01
-.49613531169529E-01 .30748993469716E-02 .19335467202533E-02 .85320645905588E-03 .78690221589046E-01
-.50051513532798E-01 .24843979837337E-02 .14120133532538E-02 .40264520065144E-03 .78531100243467E-01
-.11202822487627E-01 .14676039471344E-01 .13893139793508E-01 .13174723471373E-01 .52167493728903E-01


-.31270146475486E-01-.61104461249715E-01-.59131530183871E-01-.25013964552678E-01 .92762427753814E-02
 .56249348874907E-02 .97804294854293E-02 .11471236315805E-01-.20727131162579E-01-.20541965411089E-01
 .84022932127701E-02 .12713525336701E-01 .14350815038782E-01 .15952311038522E-01 .15776677375776E-01
 .11133006149244E-01 .15609595065236E-01 .17203577711480E-01 .18772625921528E-01 .18356824528410E-01
 .13826766612019E-01 .18478933637849E-01 .20040342450109E-01 .21588086253953E-01 .20943549621534E-01
 .47818393763294E-01 .84200415473009E-01 .85959404535894E-01 .87716287349946E-01 .55311079804659E-01
MAX ERROR AND U AND V POWER  .31432E-05
NO OF ITERATIONS ARE  11.000




 .00000000000000      .00000000000000      .00000000000000      .00000000000000      .00000000000000
 .00000000000000      .55123024654173E-03 .82694462070983E-03 .12280766662105E-02 .00000000000000
 .00000000000000      .39393391967157E-03 .49995519458721E-03 .29665774727785E-03 .00000000000000
 .00000000000000      .19239739927943E-03 .27581610960687E-03 .19194841853472E-03 .00000000000000
 .00000000000000     -.14394697336957E-05 .12611643083897E-03 .19914970340528E-03 .00000000000000
 .00000000000000      .00000000000000      .00000000000000      .00000000000000      .00000000000000


 .00000000000000      .00000000000000      .00000000000000      .00000000000000      .00000000000000
 .00000000000000      .64468343209846E-03 .70332404696872E-03-.93075828895523E-04 .00000000000000
 .00000000000000      .95417502989914E-03 .11875137686298E-02 .84244398417197E-03 .00000000000000
 .00000000000000      .10528211003075E-02 .14008863763407E-02 .11776913525051E-02 .00000000000000
 .00000000000000      .79048475252198E-03 .10379124877481E-02 .93501620599902E-03 .00000000000000
 .00000000000000      .00000000000000      .00000000000000      .00000000000000      .00000000000000




MAX ERROR AND U AND V POWER  .23423E-05
NO OF ITERATIONS ARE  11.000
MAX ERROR AND U AND V POWER  .40834E-05
NO OF ITERATIONS ARE  8.0000
TIME STEP AND TOTAL THICKNESS ARE   1.00000000000        3.29669230773
TIME STEP AND TOTAL THICKNESS ARE   1.00000000000        62.6692788514
```

```
OUTFLOW AND NET OUTFLOW ARE  .9147826E-04  .9147826E-04


 .00000000000000      00000000000000     .00000000000000     .00000000000000     .00000000000000
 .00000000000000     .41057247550924E-04 .56846035029838E-04 .81468058298827E-04 .00000000000000
 .00000000000000     .28138928200678E-04 .33823664003268E-04 .18482411011226E-04 .00000000000000
 .00000000000000     .12451366106910E-04 .18646878541670E-04 .13083451735438E-04 .00000000000000
 .00000000000000    -.17442721549091E-05 .93035439167671E-05 .14553120213194E-04 .00000000000000
  00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000


 .00000000000000     .00000000000000     .00000000000000     .00000000000000      00000000000000
 .00000000000000     .54421502682481E-04 .61766223432941E-04 .13906653062935E-05 .00000000000000
 .00000000000000     .82072462056398E-04 .10502405617843E-03 .74681432747438E-04 .00000000000000
 .00000000000000     .90320025744148E-04 .12079027872319E-03 .10016494703124E-03 .00000000000000
 .00000000000000     .66972170170303E-04 .89820544754459E-04 .79701598118132E-04 .00000000000000
  00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000


 .00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000
 .00000000000000     3.2982791312153     3.2982375791387     3.2982877200253     .00000000000000
 .00000000000000     3.2982775716448     3.2982826576149     3.2982446106205     3.2984181030719
 .00000000000000     3.2983322686140     3.2983470141199     3.2983647286222     3.2983948719993
 .00000000000000     3.2984023133110     3.2984301493117     3.2984468665240     3.2984427139725
 .00000000000000     3.2984662040978     3.2985539676780     3.2985750780206     3.2984953018081


 .00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000
 .00000000000000     .99996700313530     .99718959907456     1.0000000000000     .00000000000000
 .00000000000000     .99996653006558     .99996807280986     .99442693180934     1.0000000000000
 .00000000000000     .99998312147957     .99998759428303     .99999296768207     .99723731117466
 .00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000
 .00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000


 .00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000
 .00000000000000     4120879120879.1     4120879120879.1     4120879120879.1     .00000000000000
 .00000000000000     4120879120879.1     4120879120879.1     3910418581531.3     4120879120879.1
 .00000000000000     4120879120879.1     4120879120879.1     4120879120879.1     4120879120879.1
 .00000000000000     4120879120879.1     4120879120879.1     4120879120879.1     4120879120879.1
 .00000000000000     4120879120879.1     4120879120879.1     3630562648686.7     4120879120879.1


 .00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000
 .00000000000000     .16846153846154E-02 .16846153846154E-02 .16846153846154E-02 .00000000000000
 .00000000000000     .16846153846154E-02 .16846153846154E-02 .16846153846154E-02 .16846153846154E-02
  00000000000000     .16846153846154E-02 .16846153846154E-02 .16846153846154E-02 .16846153846154E-02
 .00000000000000     .16846153846154E-02 .16846153846154E-02 .16846153846154E-02 .16846153846154E-02
 .00000000000000     .16846153846154E-02 .16846153846154E-02 .16846153846154E-02 .16846153846154E-02

  00000000000000     .00000000000000     .00000000000000     .00000000000000     . .00000000000000
 .00000000000000    -3941.0269520577    -2998.2341796075    -4615.0152605708    -6808461394898.5
  00000000000000    -3338.6697984968    -4740.3650366613    -2960.4275986860    -11643.622254355
 .00000000000000    -5745.5721754968    -7692.1710279809    -8373.7826942921    -9389.7083890993
  00000000000000    -7943.2686226044    -8559.3741244123    -9082.1036741092    -10038.926290415
  00000000000000    -10397.702305097    -13449.670173992    -13633.964519457    -11813.305386052


 .13993055555556E-05 .22569444444444E-06 .53240740740741E-07 .42824074074074E-07 .35879629629630E-07
 .31250000000000E-07 .24305555555556E-07 .16203703703704E-07 .10416666666667E-07 .34722222222222E-08
 .00000000000000    -.23148148148148E-08-.34722222222222E-08-.46296296296296E-08-.57870370370370E-08
-.57870370370370E-08-.69444444444444E-08-.69444444444444E-08-.69444444444444E-08-.69444444444444E-08
-.69444444444444E-08


SQUARE VELOCITY,SQ. VELOCITY DIFFERENCE,MAX CHANGE
   .966152836119E-07   .119455001931E-04   .128009609762E-02
```

```
TIME STEP AND TOTAL THICKNESS ARE   1.00000000000        62.6692788514
OUTFLOW AND NET OUTFLOW ARE  .9147826E-04  .9147826E-04
GROWTH AND NET GROWTH ARE  .3200769E-01  .3200769E-01
OPEN WATER GROWTH AND NET GROWTH ETC   .0000000        .0000000       18.98872       18.98872
MAX ERROR AND U AND V POWER  .41442E-05
NO OF ITERATIONS ARE  2.0000
MAX ERROR AND U AND V POWER  .24269E-05
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   2.00000000000        62.7024471339
OUTFLOW AND NET OUTFLOW ARE  .8751464E-04  .1789929E-03
GROWTH AND NET GROWTH ARE  .3325580E-01  .6526349E-01
OPEN WATER GROWTH AND NET GROWTH ETC   .1363857E-02   .1363857E-02   18.99109       37.97980
MAX ERROR AND U AND V POWER  .28397E-05
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .13899E-05
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   3.00000000000        62.7353028718
OUTFLOW AND NET OUTFLOW ARE  .8244857E-04  .2614415E-03
GROWTH AND NET GROWTH ARE  .3293819E-01  .9820168E-01
OPEN WATER GROWTH AND NET GROWTH ETC   .1077774E-02   .2441631E-02   18.99294       56.97274
MAX ERROR AND U AND V POWER  .96390E-06
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .88624E-06
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   4.00000000000        62.7679033279
OUTFLOW AND NET OUTFLOW ARE  .7928826E-04  .3407297E-03
GROWTH AND NET GROWTH ARE  .3267974E-01  .1308814
OPEN WATER GROWTH AND NET GROWTH ETC   .8536515E-03   .3295283E-02   18.99438       75.96712
MAX ERROR AND U AND V POWER  .66153E-06
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .67121E-06
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   5.00000000000        62.8002938478
OUTFLOW AND NET OUTFLOW ARE  .7855221E-04  .4192819E--03
GROWTH AND NET GROWTH ARE  .3246907E-01  .1633505
OPEN WATER GROWTH AND NET GROWTH ETC   .6795474E-03   .3974830E-02   18.99549       94.96261
MAX ERROR AND U AND V POWER  .46274E-06
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .45260E-06
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   6.00000000000        62.8325112472
OUTFLOW AND NET OUTFLOW ARE  .7921190E-04  .4984938E-03
GROWTH AND NET GROWTH ARE  .3229661E-01  .1956471
OPEN WATER GROWTH AND NET GROWTH ETC   .5454454E-03   .4520276E-02   18.99634       113.9590
MAX ERROR AND U AND V POWER  .36399E-06
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .31219E-06
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   7.00000000000        62.8645845271
OUTFLOW AND NET OUTFLOW ARE  .8060446E-04  .5790983E-03
GROWTH AND NET GROWTH ARE  .3215388E-01  .2278010
OPEN WATER GROWTH AND NET GROWTH ETC   .4424561E-03   .4962732E-02   18.99699       132.9559
MAX ERROR AND U AND V POWER  .28442E-06
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .21506E-06
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   8.00000000000        62.8965363328
OUTFLOW AND NET OUTFLOW ARE  .8238654E-04  .6614848E-03
GROWTH AND NET GROWTH ARE  .3203419E-01  .2598352
OPEN WATER GROWTH AND NET GROWTH ETC   .3635552E-03   .5326287E-02   18.99748       151.9534
MAX ERROR AND U AND V POWER  .22296E-06
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .14870E-06
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   9.00000000000        62.9283854123
OUTFLOW AND NET OUTFLOW ARE  .8433947E-04  .7458243E-03
GROWTH AND NET GROWTH ARE  .3193342E-01  .2917686
OPEN WATER GROWTH AND NET GROWTH ETC   .3044336E-03   .5630721E-02   18.99785       170.9513
MAX ERROR AND U AND V POWER  .17618E-06
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .10368E-06
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   10.0000000000        62.9601454288
OUTFLOW AND NET OUTFLOW ARE  .8631673E-04  .8321410E-03
GROWTH AND NET GROWTH ARE  .3184633E-01  .3236149
OPEN WATER GROWTH AND NET GROWTH ETC   .2596003E-03   .5890321E-02   18.99813       189.9494
MAX ERROR AND U AND V POWER  .14327E-06
```

```
NO OF ITERATIONS ARE  1 0000
MAX ERROR AND U AND V POWER  .74802E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   11.0000000000        62.9918267462
OUTFLOW AND NET OUTFLOW ARE  .8824076E-04  .9203818E-03
GROWTH AND NET GROWTH ARE  .3176956E-01  .3553845
OPEN WATER GROWTH AND NET GROWTH ETC   .2255131E-03   .6115834E-02   18.99835        208.9478
MAX ERROR AND U AND V POWER  .11812E-06
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .60845E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   12.0000000000        63.0234373316
OUTFLOW AND NET OUTFLOW ARE  .9006798E-04  .1010450E-02
GROWTH AND NET GROWTH ARE  .3170065E-01  .3870851
OPEN WATER GROWTH AND NET GROWTH ETC   .1996125E-03   .6315447E-02   18.99851        227.9463
MAX ERROR AND U AND V POWER  .99636E-07
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .53976E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   13.0000000000        63.0549833090
OUTFLOW AND NET OUTFLOW ARE  .9177519E-04  .1102225E-02
GROWTH AND NET GROWTH ARE  .3163775E-01  .4187229
OPEN WATER GROWTH AND NET GROWTH ETC   .1799415E-03   .6495388E-02   18 99864        246 9449
MAX ERROR AND U AND V POWER  .85949E-07
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .49112E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   14.0000000000        63.0864693874
OUTFLOW AND NET OUTFLOW ARE  .9335328E-04  .1195578E-02
GROWTH AND NET GROWTH ARE  .3157943E-01  .4503023
OPEN WATER GROWTH AND NET GROWTH ETC   1650067E-03    6660395E-02   18 99873        265.9436
MAX ERROR AND U AND V POWER  .80681E-07
NO OF ITERATIONS ARE  1 0000
MAX ERROR AND U AND V POWER  .45665E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   15.0000000000        63.1178991909
OUTFLOW AND NET OUTFLOW ARE  .9480144E-04  .1290380E-02
GROWTH AND NET GROWTH ARE  .3152460E-01  .4818269
OPEN WATER GROWTH AND NET GROWTH ETC   .1536697E-03   .6814064E-02   18 99880        284.9424
MAX ERROR AND U AND V POWER  .76632E-07
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .43203E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   16.0000000000        63.1492755131
OUTFLOW AND NET OUTFLOW ARE  .9612368E-04  .1386503E-02
GROWTH AND NET GROWTH ARE  .3147245E-01  .5132994
OPEN WATER GROWTH AND NET GROWTH ETC   .1450639E-03   .6959128E-02   18 99885        303.9413
MAX ERROR AND U AND V POWER  .73288E-07
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .41433E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   17.0000000000        63.1806005122
OUTFLOW AND NET OUTFLOW ARE  .9732675E-04  .1483830E-02
GROWTH AND NET GROWTH ARE  .3142233E-01  .5447217
OPEN WATER GROWTH AND NET GROWTH ETC   .1385303E-03   .7097658E-02   18.99889        322.9402
MAX ERROR AND U AND V POWER  .69765E-07
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .40019E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   18.0000000000        63.2118759303
OUTFLOW AND NET OUTFLOW ARE  .9842014E-04  .1582250E-02
GROWTH AND NET GROWTH ARE  .3137384E-01  .5760955
OPEN WATER GROWTH AND NET GROWTH ETC   .1336429E-03   .7231301E-02   18.99892        341.9391
MAX ERROR AND U AND V POWER  .67617E-07
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .38332E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   19.0000000000        63.2431035050
OUTFLOW AND NET OUTFLOW ARE  .9942083E-04  .1681671E-02
GROWTH AND NET GROWTH ARE  .3132700E-01  .6074225
OPEN WATER GROWTH AND NET GROWTH ETC   .1304179E-03   .7361719E-02   18.99894        360,9381
MAX ERROR AND U AND V POWER  .67242E-07
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER  .38222E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   20.0000000000        63.2742841446
OUTFLOW AND NET OUTFLOW ARE  .1003302E-03  .1782001E-02
```

```
GROWTH AND NET GROWTH ARE  .3128097E-01   .6387035
OPEN WATER GROWTH AND NET GROWTH ETC    1279838E-03    7489703E-02   18 99896        379.9370
MAX ERROR AND U AND V POWER  .66773E-07
NO OF ITERATIONS ARE  1.0000
MAX ERROR AND U AND V POWER   36318E-07
NO OF ITERATIONS ARE  1.0000
TIME STEP AND TOTAL THICKNESS ARE   21.0000000000        3 32958600935
TIME STEP AND TOTAL THICKNESS ARE   21 0000000000        63.3054187372


OUTFLOW AND NET OUTFLOW ARE  . :)11530E-03  .1883154E-02


 .00000000000000     .00000000000000     .00000000000000     00000000000000      00000000000000
 .00000000000000     .39112283619156E-04 .55390625151813E-04 .79737015294279E-04 .00000000000000
 .00000000000000     .26984604779460E-04 .32769926926668E-04 .17828042226492E-04 .00000000000000
 .00000000000000     .11077839177733E-04 .16274783134873E-04 .11429718928289E-04 .00000000000000
 .00000000000000     .33472924772821E-05 .62165604549794E-05 .13457354034109E-04 .00000000000000
 .00000000000000     .00000000000000     00000000000000      .00000000000000     0000000000000


 .00000000000000     .00000000000000     00000000000000      00000000000000      .00000000000000
 00000000000000      .52957772499348E-04 .59851900358741E-04 .38156306695244E-06 .00000000000000
 .00000000000000     .79224453328352E-04 .10036910526617E-03 .72011051835235E-04 .00000000000000
 .00000000000000     .86993872316885E-04 .11613710406754E-03 .96855984472808E-04 .00000000000000
 .00000000000000     .65075272625387E-04 .85091061307831E-04 .75753949834686E-04 .00000000000000
 .00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000


 .00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000
 .00000000000000     3.3297049985766     3.3297683558338     3.3303937609084     .00000000000000
 .00000000000000     3.3296992766615     3.3301275693885     3.3307122694294     3.3324164516894
 .00000000000000     3.3305525598485     3.3309494408961     3.3316519560007     3.3325288078242
 .00000000000000     3.3319192285176     3.3325344550561     3.3329076517258     3.3328853634415
 .00000000000000     3.3332875151558     3.3346405948677     3.3349120908281     3.3338263905225


 .00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000
 .00000000000000     .99986326260534     .99981699369587     .99986487686175     .00000000000000
 .00000000000000     .99986991657685     .99986419827627     .99984134288016     1.0000000000000
 .00000000000000     .99993553914699     .99994836046999     .99996066602523     1.0000000000000
 .00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000
 .00000000000000     1.0000000000000     1.0000000000000     1.0000000000000     1.0000000000000


 .00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000
 .00000000000000     4148797395966.0     4144960670301.6     4149649203475.0     .00000000000000
 .00000000000000     4149350950360.9     4149322178617.5     4148079023280.0     4163479172659.7
 .00000000000000     4155835164294.7     4157343734269.7     4159156471236.6     4163612259607.1
 .00000000000000     4162827530141.9     4163558776885.6     4164004533269.4     4163986334457.3
 .00000000000000     4164457586814.3     4166088094417.2     4166413006223.2     4165099975715.7


 .00000000000000     .00000000000000     .00000000000000     .00000000000000     .00000000000000
 .00000000000000     .16563027461489E-02 .16616329270765E-02 .16553444907767E-02 .00000000000000
 .00000000000000     .16555319122144E-02 .16557128568464E-02 .16576196172958E-02 .16369033266212E-02
 .00000000000000     .16468472179467E-02 16448891835361E-02  .16426152050027E-02 .16367542692401E-02
 .00000000000000     .16376331662411E-02 .16368141698881E-02 .16363149227383E-02 .16363353054078E-02
 .00000000000000,    .16358075027679E-02 .16339813342527E-02 .16336174330300E-02 .16350880271984E-02


 .00000000000000     ,00000000000000     .00000000000000     .00000000000000     .00000000000000
 .00000000000000     -6826.6410687273    -6549.0875390725    -7044.6807632523    -9957.9943333619
 .00000000000000     -6599.8202405354    -7008.8109149810    -6706.8246676463    -9463.3352376975
 .00000000000000     -7423.2207990731    -7791.8523162416    -8224.2301610488    -8625.7248100933
 .00000000000000     -8438.4371324362    -8680.8148547810    -8931.6733844908    -9109.7450977477
 .00000000000000     -9211.6345199173    -10009.352354911    -10193.336151179    -9556.6844731167
 .13993055555556E-05 .22569444444444E-06 .53240740740741E-07 .42824074074074E-07 .35879629629630E-07
 .31250000000000E-07 .24305555555556E-07 .16203703703704E-07 .10416666666667E-07 .34722222222222E-08
 .00000000000000     -.23148148148148E-08-.34722222222222E-08-.46296296296296E-08-.57870370370370E-08
-.57870370370370E-08-.69444444444444E-08-.69444444444444E-08-.69444444444444E-08-.69444444444444E-08
-.69444444444444E-08


SQUARE VELOCITY,SQ. VELOCITY DIFFERENCE,MAX CHANGE
   .893987030642E-07   .658437122999E-13   .847259876693E-07


TIME STEP AND TOTAL THICKNESS ARE   21.0000000000        63.3054187372
OUTFLOW AND NET OUTFLOW ARE  .1011530E-03  .1883154E-02
GROWTH AND NET GROWTH ARE  .3123575E-01  .6699393
OPEN WATER GROWTH AND NET GROWTH ETC   .1263262E-03   .7616029E-02   18.99897        398.9360

14.277 sec.   501 I/O
```

☆U.S. GOVERNMENT PRINTING OFFICE: 1980 4 600-265/401

ILMED

8